

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平10-155149

(43)公開日 平成10年(1998) 6 月 9 日

(51)Int.Cl.<sup>6</sup>

識別記号

F I

H 0 4 N 7/30

H 0 4 N 7/133

Z

H 0 4 L 29/08

7/15

H 0 4 N 7/15

H 0 4 L 13/00

3 0 7 C

審査請求 未請求 請求項の数7 O L (全 37 頁)

(21)出願番号

特願平8-313304

(22)出願日

平成8年(1996)11月25日

(71)出願人 000005821

松下電器産業株式会社

大阪府門真市大字門真1006番地

(72)発明者 荒瀬 吉隆

広島県広島市東区光町1丁目12番20号 株

式会社松下電器情報システム広島研究所内

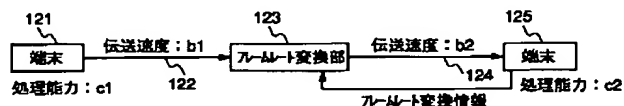
(74)代理人 弁理士 早瀬 憲一

(54)【発明の名称】 フレームレート変換方式、及びフレームレート変換装置

(57)【要約】

【課題】 端末の処理能力に合わせたフレームレート変換を行うこと、および動き補償を伴うブロックを含むハイブリッド符号化された画像データの高速で負荷の少ないフレームレート変換を行うこと。

【解決手段】 入力された動き補償を伴うブロックを含むハイブリッド符号化された画像データを、逆直交変換、動き検出、直交変換を行わず、複数枚のフレームを1枚に合成して、端末の処理能力に合わせたフレーム数の動き補償を伴わないブロックとフレーム内符号化されたブロックのハイブリッド符号化された画像データに変換し出力する。





## 【特許請求の範囲】

【請求項 1】 端末間でハイブリッド符号化された画像データを通信する場合において、一方の端末を a 1、他方の端末を a 2 とし、端末 a 1 が接続されている通信回線の伝送速度を b 1、端末 a 2 が接続されている通信回線の伝送速度を b 2、さらに、端末 a 1 の処理能力を c 1、端末 a 2 の処理能力を c 2 としたとき、

フレームレートの高い画像データを端末 a 1 側から入力し、端末 a 2 側に出力するフレーム数を、伝送速度が  $b_1 \geq b_2$  で、かつ端末の処理能力が  $c_1 = c_2$  のときには、 $b_2$  で伝送できるフレーム数に設定し、伝送速度が  $b_1 \geq b_2$  で、かつ端末の処理能力が  $c_1 > c_2$  のときには、 $b_2$  で伝送できるフレーム数と  $c_2$  で処理できるフレーム数のどちらか少ない方のフレーム数に設定し、伝送速度が  $b_1 < b_2$  で、かつ端末の処理能力が  $c_1 > c_2$  のときには、 $b_1$  で伝送できるフレーム数と  $c_2$  で処理できるフレーム数のどちらか少ない方のフレーム数に設定し、伝送速度が  $b_1 > b_2$  で、かつ端末の処理能力が  $c_1 < c_2$  のときには、 $b_2$  で伝送できるフレーム数と  $c_1$  で処理できるフレーム数のどちらか少ない方のフレーム数に設定し、上記端末 a 1 から入力されたフレームレートの高い画像データのフレームレートを変換して上記端末 a 2 側にフレームレートの低い画像データとして出力することを特徴とするフレームレート変換方式。

【請求項 2】 フレームレートの高い動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データを入力し、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データのフレームを複数枚まとめて 1 枚の動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データのフレームに合成することによりフレームレートを変換し、フレームレートの低い動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データとして出力することを特徴とするフレームレート変換方式。

【請求項 3】 フレームレートの高い動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データとの両方が含まれる画像データを入力し、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データとの両方が含まれる画像データのフレームを複数枚まとめて 1 枚の動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データのフレームに合成することによりフレームレートを変換し、フレ

ームレートの低い動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データとして出力することを特徴とするフレームレート変換方式。

【請求項 4】 フレームレートの高い動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データを入力し、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データのフレームを複数枚まとめて 1 枚の動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データのフレームに合成することによりフレームレートを変換し、フレームレートの低い動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データとして出力することを特徴とするフレームレート変換方式。

【請求項 5】 フレームレートの高い動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの 3 つが含まれる画像データを入力し、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの 3 つが含まれる画像データのフレームを複数枚まとめて 1 枚の動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データのフレームに合成することによりフレームレートを変換し、フレームレートの低い動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データとして出力することを特徴とするフレームレート変換方式。

【請求項 6】 請求項 1 記載のフレームレート変換方式において、フレームレートの変換を行う際に、上記請求項 2 ないし請求項 5 のいずれかに記載のフレームレート変換方式を用いてフレームレート変換を行うことを特徴とするフレームレート変換方式。

【請求項 7】 上記請求項 1 ないし請求項 6 のいずれかに記載のフレームレート変換方式を用いてフレームレー



トの変換を行うことを特徴とするフレームレート変換装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、ビデオ会議や動画伝送を行うにあたり、勧告H. 261などを用い、処理能力の異なる端末間でハイブリッド符号化された画像データを伝送する際、または伝送速度の異なる通信回線を中継しハイブリッド符号化された画像データを伝送する際、または処理能力が異なる端末間で伝送速度の異なる通信回線を中継しハイブリッド符号化された画像データを伝送する際のフレームレート変換方式、および動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータが含まれる画像データのフレームレート変換方式、および動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータが含まれる画像データのフレームレート変換方式を利用した処理能力の異なる端末間でハイブリッド符号化された画像データを伝送する際、または伝送速度の異なる通信回線を中継しハイブリッド符号化された画像データを伝送する際、または処理能力が異なる端末間で、伝送速度の異なる通信回線を中継しハイブリッド符号化された画像データを伝送する際のフレームレート変換方式、およびフレームレート変換装置に関するものである。

【0002】

【従来の技術】 従来のフレームレート変換装置について図面を参照しながら説明する。図20は、従来のフレームレート変換方式（装置）の構成を示すブロック図である。図20において、821、825は端末、822は端末821に接続されている、伝送速度の速い通信回線、824は端末825に接続されている、伝送速度の遅い通信回線、823は、上記伝送速度の速い通信回線822と上記伝送速度の遅い通信回線824との間に設けられた従来のフレームレート変換装置であるフレームレート変換部である。

【0003】 次に動作について説明する。図20において、フレームレート変換装置は、端末821から出力されたフレームレートの高いハイブリッド符号化された画像データを実時間で端末825に伝送するために、フレームレート変換部823は、速い通信回線822上の、フレームレートの高いハイブリッド符号化された画像データをフレームレート変換し、遅い通信回線824上にフレームレートの低いハイブリッド符号化された画像データとして出力していた。

【0004】 また、動き補償を伴うフレーム間予測誤差に対して、直交変換を行うハイブリッド符号化された画像データのフレームレート変換方法として、例えば、特開平7-50834号公報に示されるレート変換画像符号化装置において、可変長復号器、及び可変長符号器を

追加した構成の装置によるフレームレート変換方法を例に挙げて、以下、図面を参照しながら説明する。

【0005】 図10は上記特開平7-50834号公報のレート変換画像符号化装置において可変長復号器、可変長符号器を追加した構成のレート変換画像符号化装置を示すものである。

【0006】 図10において、901は画像データの入力、902は入力された画像データを復号化する可変長復号化手段、903は復号化されたデータを逆量子化する逆量子化手段、919は逆量子化手段903の出力と画像メモリ920から出力されたデータとを加算する加算手段、921、922はセクタ、923は逆直交変換手段、924は加算手段、925は画像メモリ、926は画像メモリ925から出力される画像メモリを入力とするループ内フィルタ手段、927はセクタ、928は減算手段、930は直交変換手段、931はセクタ、914は量子化手段、932は逆量子化手段、933は逆直交変換手段、934は加算手段、935は画像メモリ、936はループ内フィルタ手段、937はセクタ、938は動き検出手段、939は符号化制御手段、917は可変長符号化手段、918は出力である。

【0007】 次に動作について説明する。図10において、入力901より入力された、フレームレートの高い動き補償を伴うフレームのレート変換を行うには、まず、ハイブリッド符号化された画像データを、可変長復号化手段902により可変長復号化し、量子化された動き補償を伴う予測誤差直交変換データに変換し、続いて、逆量子化手段903によって逆量子化して動き補償を伴う予測誤差直交変換データに変換する。

【0008】 そして、上記予測誤差直交変換データは、セクタ922により選択され、逆直交変換手段923により逆直交変換されて動き補償を伴う予測誤差データに変換され、加算手段924において、セクタ927にて選択されたループ内フィルタ手段926を経由した画像メモリ925と加算されて係数（画素データ）に変換される。

【0009】 そして、加算手段924より出力された係数（画素データ）と画像メモリ935上の係数（画素データ）とを用いて、動き検出手段938は、動き検出を行って動きベクトルを求め、減算手段928により、加算手段924より出力された係数（画素データ）から、動きベクトルを考慮した画像メモリ935をループ内フィルタ手段936を経由させた後に減算し、動き補償を伴う予測誤差データに変換する。

【0010】 そして、上記得られた動き補償を伴う予測誤差データは、直交変換手段930により直交変換されて、動き補償を伴う予測誤差直交変換データに変換され、セクタ931により選択され、後段の量子化手段914により量子化されて、量子化された動き補償を伴う予測誤差直交変換データに変換され、続いて、可変長



符号化手段917により可変長符号化され、ハイブリッド符号化された画像データに変換され、出力918よりフレームレートの低い動き補償を伴うフレームとして出力される。

【0011】また、フレームレートの高いハイブリッド符号化された画像データがフレーム内符号化された画像データのみであれば、逆直交変換、動き検出、直交変換などの途中の処理を省くことができるが、動き補償を伴うブロックが1つでも含まれていれば、動きベクトルを検出する必要が生じるため、上記の処理を全て行う必要があった。

【0012】上述のように構成されたフレームレート変換装置では、フレームレートを変換する際に、通信回線の伝送速度を基準として変換を行っていた。すなわち、図20において、例えば、端末821が30fpsのエンコード能力を有し、端末825が5fpsのデコード能力を有し、通信回線822が30fpsの伝送能力を有し、通信回線824が15fpsの伝送能力を有するとする。この時、フレームレート変換装置(変換部823)では、30fpsのフレームを15fpsになるよう

に変換していた。

【0013】このため、通信回線824上には15fpsのハイブリッド符号化された画像データが伝送されることになるが、端末825は5fpsの速度でしかデコードできないために、5fpsの速度でしか画像を表示することができない。このことから、デコードされない不要なハイブリッド符号化された画像データが通信回線824上を伝送されていたことが分かる。従って、フレームレート変換部823の後段の通信回線824の使用効率が悪くなる。

【0014】さらに、上記から分かるように、動き補償を伴うフレーム、および動き補償を伴うブロックを含むフレームのレート変換を行うためには、ハイブリッド符号化されたデータの復号化と、係数(画素データ)の符号化のための全てのプロセスを実行する必要があり、変換に時間と負荷がかかっていた。

【0015】

【発明が解決しようとする課題】従来のフレームレート変換装置は以上のように構成されており、伝送速度の異なる通信回線間において、単に、伝送速度の遅い方の通信回線に合わせて、これに適したフレームレートとなるように変換するようにしていたために、フレームレート変換装置よりも後段の通信回線の使用効率が悪化することがあるという問題点があった。また、動き補償を伴うフレームを処理する際の処理時間が長く、かつ、システムにも大きな負荷がかかるという問題点があった。

【0016】本発明は以上のような問題点を解消するためになされたもので、フレームレートの高いハイブリッド符号化された画像データを、フレームレートの低いハイブリッド符号化された画像データに変換する際、端末

の処理能力を考慮することにより、通信回線の使用効率の向上と、端末の処理能力に対応したフレームレートのハイブリッド符号化された画像データを生成することができるフレームレート変換方式、及びフレームレート変換装置を提供することを目的とする。

【0017】また、動き補償を伴うフレーム、および動き補償を伴うブロックを含むフレームのレート変換を行う場合でも、演算量を大幅に減らし、高速で負荷の少ないフレームレート変換を行うことができるフレームレート変換方式、及びフレームレート変換装置を提供することを目的とする。

【0018】

【課題を解決するための手段】本発明の請求項1に係るフレームレート変換方式は、端末間でハイブリッド符号化された画像データを通信する場合において、一方の端末をa1、他方の端末をa2とし、端末a1が接続されている通信回線の伝送速度をb1、端末a2が接続されている通信回線の伝送速度をb2、さらにa1の処理能力をc1、端末a2の処理能力をc2としたとき、フレームレートの高い画像データを端末a1側から入力し、端末a2側に出力するフレーム数を、伝送速度が $b1 \geq b2$ で、かつ端末の処理能力が $c1 = c2$ のときには、b2で伝送できるフレーム数に設定し、伝送速度が $b1 \geq b2$ で、かつ端末の処理能力が $c1 > c2$ のときには、b2で伝送できるフレーム数とc2で処理できるフレーム数のどちらか少ない方のフレーム数に設定し、伝送速度が $b1 < b2$ で、かつ端末の処理能力が $c1 > c2$ のときには、b1で伝送できるフレーム数とc2で処理できるフレーム数のどちらか少ない方のフレーム数に設定し、伝送速度が $b1 > b2$ で、かつ端末の処理能力が $c1 < c2$ のときには、b2で伝送できるフレーム数とc1で処理できるフレーム数のどちらか少ない方のフレーム数に設定し、上記端末a1から入力されたフレームレートの高い画像データのフレームレートを変換して上記端末a2側にフレームレートの低い画像データとして出力するようにしたものである。

【0019】また、本発明の請求項2に係るフレームレート変換方式は、フレームレートの高い動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データを入力し、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データのフレームを複数枚まとめて1枚の動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データのフレームに合成することによりフレームレートを変換し、フレームレートの低い動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データとして出力するようにしたものである。

【0020】また、本発明の請求項3に係るフレームレート変換方式は、フレームレートの高い動き補償を伴う



フレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データとの両方が含まれる画像データを入力し、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データとの両方が含まれる画像データのフレームを複数枚まとめて1枚の動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データのフレームに合成することによりフレームレートを変換し、フレームレートの低い動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データとして出力するようにしたものである。

【0021】また、本発明の請求項4に係るフレームレート変換方式は、フレームレートの高い動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データを入力し、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データのフレームを複数枚まとめて1枚の動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データのフレームに合成することによりフレームレートを変換し、フレームレートの低い動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データとして出力するようにしたのである。

【0022】また、本発明の請求項5に係るフレームレート変換方式は、フレームレートの高い動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの3つが含まれる画像データを入力し、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの3つが含まれる画像データのフレームを複数枚まとめて1枚の動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データのフレームに合成することによりフレームレートを変換し、フレームレートの

低い動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データとして出力するようにしたものである。

【0023】本発明の請求項6に係るフレームレート変換方式は、上記請求項1記載のフレームレート変換方式において、フレームレートの変換を行う際に、上記請求項2ないし請求項5のいずれかに記載のフレームレート変換方式を用いてフレームレート変換を行うようにしたものである。

【0024】本発明の請求項7に係るフレームレート変換装置は、上記請求項1ないし請求項6のいずれかに記載のフレームレート変換方式を用いてフレームレートの変換を行うようにしたものである。

【0025】

【発明の実施の形態】

実施の形態1. 以下、本発明の請求項1に対応する実施の形態1.に係るフレームレート変換方式について、図1に示す、フレームレート変換方式の構成を参照しながら説明する。

【0026】図1はフレームレート変換方式の構成を示すブロック図である。

【0027】図1において、121は端末であり、その処理能力はc1とする。122は上記端末121の後段に接続された通信回線であり、その伝送速度はb1とする。123は上記通信回線122の後段に接続されたフレームレート変換部であり、フレームレート変換情報が端末125から入力され、通信回線122の伝送速度b1と、通信回線124の伝送速度b2と、端末125の処理能力c2とをそれぞれ考慮したフレームレート変換を行うように構成されている。124は通信回線であり、その伝送速度はb2とする。また、125は端末であり、その処理能力はc2とする。

【0028】以下、フレームレートの高い画像データを端末121側から入力し、通信回線の伝送速度が、 $b1 \geq b2$ となる関係で、かつ、端末の処理能力が、 $c1 = c2$ のときのフレームレート変換方式について図1を参照しながら説明する。

【0029】ここで、 $c1 = c2$ であるので、端末同士の処理能力は等しく、これに合わせたフレームレートの変換は考慮する必要はない。よって、通信回線間の伝送速度b1とb2の關係に着目することになる。

【0030】今、通信回線124の伝送速度b2は、通信回線122の伝送速度b1以下であるので、フレームレート変換部123では通信回線124の伝送速度b2で伝送可能なフレームレートに変換すればよいことになる。よって、端末121側から入力されたフレームレートの高いハイブリッド符号化された画像データは、伝送速度b2の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換さ



れ端末 1 2 5 側に出力される。端末 1 2 5 では、通信回線 1 2 4 の伝送速度  $b_2$  と等しいフレームレートの画像データが送られて、端末 1 2 5 の処理能力範囲で完全に再生されて表示されることになる。

【0031】次に、フレームレートの高い画像データを端末 1 2 1 側から入力し、通信回線の伝送速度が、 $b_1 \geq b_2$  となる関係で、かつ、端末の処理能力が、 $c_1 > c_2$  の関係となるときのフレームレート変換方式について図 1 を参照しながら説明する。

【0032】端末 1 2 1、1 2 5 の処理能力  $c_1$  と  $c_2$  の関係に着目すると、端末 1 2 1 の処理能力  $c_1$  よりも端末 1 2 5 の処理能力  $c_2$  が低いので、端末 1 2 5 の処理能力  $c_2$  で処理可能なフレームレートに変換すればよいことになる。また、通信回線 1 2 2、1 2 4 の伝送速度  $b_1$  と  $b_2$  の関係に着目すると、通信回線 1 2 4 の伝送速度  $b_2$  は通信回線 1 2 2 の伝送速度  $b_1$  以下であるので、通信回線 1 2 4 の伝送速度  $b_2$  で伝送可能なフレームレートに変換すればよいことになる。

【0033】ここで、端末 1 2 5 の処理能力  $c_2$  で処理可能なフレーム数よりも、通信回線 1 2 4 の伝送速度  $b_2$  で伝送可能なフレーム数の方が多い場合、端末 1 2 5 の処理能力  $c_2$  で処理可能なフレーム数を超えるフレームを送信しても端末 1 2 5 ではデコードできないので、通信回線の使用効率が悪くなることになる。これを回避するために、端末 1 2 1 側から入力されたフレームレートの高いハイブリッド符号化された画像データは、端末 1 2 5 の処理能力  $c_2$  の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末 1 2 5 側に出力される。

【0034】また、端末 1 2 5 の処理能力  $c_2$  で処理可能なフレーム数よりも、通信回線 1 2 4 の伝送速度  $b_2$  で伝送可能なフレーム数の方が少ない場合、通信回線 1 2 4 の伝送速度  $b_2$  で伝送可能なフレーム数しか伝送することができないので、端末 1 2 1 側から入力されたフレームレートの高いハイブリッド符号化された画像データは、通信回線 1 2 4 の伝送速度  $b_2$  の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末 1 2 5 側に出力される。

【0035】また、さらに端末 1 2 5 の処理能力  $c_2$  で処理可能なフレーム数と通信回線 1 2 4 の伝送速度  $b_2$  で伝送可能なフレーム数とが等しい場合、端末 1 2 1 側から入力されたフレームレートの高いハイブリッド符号化された画像データは、通信回線 1 2 4 の伝送速度  $b_2$ 、もしくは端末 1 2 5 の処理能力  $c_2$  の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末 1 2 5 側に出力される。

【0036】次に、フレームレートの高い画像データを、端末 1 2 1 側から入力し、通信回線の伝送速度が  $b_1 < b_2$  となる関係で、かつ、端末の処理能力が、 $c_1 > c_2$  の関係となるときのフレームレート変換方式について図 1 を参照しながら説明する。

1  $< b_2$  となる関係で、かつ、端末の処理能力が、 $c_1 > c_2$  の関係となるときのフレームレート変換方式について図 1 を参照しながら説明する。

【0037】端末 1 2 1、1 2 5 の処理能力  $c_1$  と  $c_2$  の関係に着目すると、端末 1 2 1 の処理能力  $c_1$  よりも端末 1 2 5 の処理能力  $c_2$  が低いので、端末 1 2 5 の処理能力  $c_2$  で処理可能なフレームレートに変換すればよいことになる。また、通信回線 1 2 2、1 2 4 の伝送速度  $b_1$  と  $b_2$  の関係に着目すると、通信回線 1 2 2 の伝送速度  $b_1$  は通信回線 1 2 4 の伝送速度  $b_2$  よりも遅いので、伝送速度  $b_1$  で伝送可能なフレームレートに変換すればよいことになる。

【0038】ここで、端末 1 2 5 の処理能力  $c_2$  で処理可能なフレーム数よりも通信回線 1 2 2 の伝送速度  $b_1$  で伝送可能なフレーム数の方が多い場合、端末 1 2 5 の処理能力  $c_2$  で処理可能なフレーム数を超えるフレームを送信しても端末 1 2 5 ではデコードできないので、通信回線の使用効率が悪くなることになる。これを回避するために、端末 1 2 1 側から入力されたフレームレートの高いハイブリッド符号化された画像データは、端末 1 2 5 の処理能力  $c_2$  の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末 1 2 5 側に出力される。

【0039】また、端末 1 2 5 の処理能力  $c_2$  で処理可能なフレーム数よりも、通信回線 1 2 2 の伝送速度  $b_1$  で伝送可能なフレーム数の方が少ない場合、通信回線 1 2 2 の伝送速度  $b_1$  で伝送可能なフレーム数しか伝送することができないので、端末 1 2 1 側から入力されたフレームレートの高いハイブリッド符号化された画像データは、通信回線 1 2 2 の伝送速度  $b_1$  の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末 1 2 5 側に出力される。

【0040】また、さらに、端末 1 2 5 の処理能力  $c_2$  で処理可能なフレーム数と通信回線 1 2 2 の伝送速度  $b_1$  で伝送可能なフレーム数とが等しい場合、端末 1 2 1 側から入力されたフレームレートの高いハイブリッド符号化された画像データは、通信回線 1 2 2 の伝送速度  $b_1$ 、もしくは端末 1 2 5 の処理能力  $c_2$  の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末 1 2 5 側に出力される。

【0041】次に、フレームレートの高い画像データを端末 1 2 1 側から入力し、通信回線の伝送速度が  $b_1 > b_2$  となる関係で、かつ、端末の処理能力が、 $c_1 < c_2$  の関係となるときのフレームレート変換方式について図 1 を参照しながら説明する。

【0042】端末 1 2 1、1 2 5 の処理能力  $c_1$  と  $c_2$  の関係に着目すると、端末 1 2 5 の処理能力  $c_2$  よりも端末 1 2 1 の処理能力  $c_1$  が低いので、端末 1 2 1 の処



理能力 $c_1$ で処理可能なフレームレートに変換すればよいことになる。また、通信回線122、124のの伝送速度 $b_1$ と $b_2$ の關係に着目すると、通信回線124の伝送速度 $b_2$ は通信回線122の伝送速度 $b_1$ よりも遅いので、通信回線124の伝送速度 $b_2$ で伝送可能なフレームレートに変換すればよいことになる。

【0043】ここで、端末121の処理能力 $c_1$ で処理可能なフレーム数よりも、通信回線124の伝送速度 $b_2$ で伝送可能なフレーム数の方が多い場合、端末121の処理能力 $c_1$ で処理可能なフレーム数を超えるフレームは通信回線124の伝送速度 $b_2$ を利用しても、伝送できない。よって、端末121側から入力されたハイブリッド符号化された画像データは、フレームレート変換されずに、そのまま端末125側に出力される。

【0044】また、端末121の処理能力 $c_1$ で処理可能なフレーム数よりも、通信回線124の伝送速度 $b_2$ で伝送可能なフレーム数の方が少ない場合、通信回線124の伝送速度 $b_2$ で伝送可能なフレーム数しか伝送することができないので、端末121側から入力されたフレームレートの高いハイブリッド符号化された画像データは、通信回線124の伝送速度 $b_2$ の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末125側に出力される。

【0045】また、端末121の処理能力 $c_1$ で処理可能なフレーム数と通信回線124の伝送速度 $b_2$ で伝送可能なフレーム数とが等しい場合、端末121側から入力されたハイブリッド符号化された画像データは、フレームレート変換されずに、そのまま端末125側に出力される。

【0046】以下、フレームレート変換処理の詳細について説明する。入力される画像データを、0から $i$ 番目のフレームまではフレームレート変換なしとし、 $(i+1)$ から $(i+n)$ 番目のフレームまでを合成してフレームレート変換を行い、 $(i+n+1)$ 番目のフレーム以降のフレームは、フレームレート変換なしとして説明する。また、入力されたフレームレートの高い画像データがフレーム内符号化されたデータである場合を例にあげて、図面を参照しながら説明する。

【0047】図2は入力されたフレームレートの高い画像データのうち、 $(i+1)$ から $(i+n)$ 番目のフレームまでが、フレーム内符号化されたデータである場合の、 $(i+n)$ 番目のフレームの処理時のフレームレート変換方式を示すものである。

【0048】図2において、101は入力であり、フレームレートの高い画像データが入力される。

【0049】また、102は可変長復号化手段であり、ハイブリッド符号化データを可変長復号化することにより、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換

データ、量子化された直交変換係数に変換する。また、同時に、入力側の端末の符号化装置で設定されたフレーム内/フレーム間識別を検出して、フレーム内/フレーム間( $i_n$ )を設定し、また、入力側端末の符号化装置で設定された、動き補償あり/なし識別を検出して、動き補償あり/なし( $i_n$ )を設定し、また、入力側端末の符号化装置で設定された量子化係数を検出して、量子化係数( $i_n$ )を設定し、さらに、入力側端末の符号化装置で設定された動きベクトルを検出して、動きベクトル( $i_n$ )を設定する。

【0050】103は逆量子化手段であり、上記設定された量子化係数( $i_n$ )により、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を、動き補償を伴う予測誤差直交変換データ、動き補償を伴わない予測誤差直交変換データ、直交変換係数にそれぞれ変換する。

【0051】104は予測誤差直交変換データ加算手段であり、動きベクトル・直交変換係数演算手段108にて動きベクトルを考慮して演算された直交変換係数、または画像メモリ106に記憶されている直交変換係数と、動き補償を伴う予測誤差直交変換データ、または動き補償を伴わない予測誤差直交変換データとを加算する。

【0052】105は直交変換係数選択手段であり、上記設定されたフレーム内/フレーム間( $i_n$ )により制御される。すなわち、上記フレーム内/フレーム間( $i_n$ )が「フレーム内」の時は、逆量子化手段103により逆量子化された直交変換係数が選択され、上記フレーム内/フレーム間( $i_n$ )が「フレーム間」の時は、予測誤差直交変換データ加算手段104により算出された直交変換係数が選択され出力される。

【0053】106は画像メモリであり、上記直交変換係数選択手段105により選択された直交変換係数を記憶するためのものである。

【0054】107は動き補償あり/なし( $i_n$ )判断手段であり、動き補償が「あり」の時は画像メモリ106に記憶されている直交変換係数を後述する動きベクトル・直交変換係数演算手段108に出力する。動き補償が「なし」の時は画像メモリ106に記憶されている直交変換係数を、予測誤差直交変換データ加算手段104に出力する。

【0055】108は動きベクトル・直交変換係数演算手段であり、動きベクトル( $i_n$ )を用いて、動き補償あり/なし( $i_n$ )判断手段107にて「あり」と判断されたときの画像メモリ106に記憶されていた直交変換係数を、マトリクス演算することにより、動きベクトルを考慮して演算された直交変換係数を算出する。マトリクス演算については後述する方法を用いることとする。

10

20

30

40

50



【0056】109は初期画像取込選択手段であり、後述する符号化制御・フレーム内／フレーム間判断手段113により設定された画像取込指示により、画像メモリ106に記憶されている直交変換係数を取り込み、後述する画像メモリ110に直交変換係数を出力する。

【0057】110は画像メモリであり、初期画像取込選択手段109により選択された直交変換係数を記憶するためのものである。

【0058】111はフレーム内／フレーム間 (in) 判断手段であり、「フレーム内」の時は、画像メモリ106の直交変換係数を後段の符号化制御・フレーム内／フレーム間判断手段113へ出力し、「フレーム間」の時は、画像メモリ106の直交変換係数を後述する直交変換係数減算手段112へ出力する。

【0059】112は直交変換係数減算手段であり、フレーム内／フレーム間 (in) 判断手段111で、「フレーム間」と判断されたときに画像メモリ106に記憶されていた直交変換係数から、画像メモリ110の直交変換係数を減算する。

【0060】113は符号化制御・フレーム内／フレーム間判断手段であり、フレームレート変換が行われていない時には、動き補償あり／なし (in) をそのまま動き補償あり／なし (out) として出力し、フレームレート変換が行われている時には、動き補償あり／なし (out) を「なし」に設定して出力する。また、フレームレート変換を行う1つ前のフレームを画像メモリ110に記憶させるために、画像取込指示を出力する。また、フレーム内／フレーム間 (in) を用いて、フレーム内／フレーム間 (in) 判断手段111からの直交変換係数と、直交変換係数減算手段112からの動き補償を伴わない予測誤差直交変換データとの選択を行い、後述する量子化手段114へ出力する。また、量子化係数 (in) を再設定し、これを量子化係数 (out) として後述する量子化手段114へ出力する。

【0061】114は量子化手段であり、上記出力された量子化係数 (out) により、動き補償を伴わない予測誤差直交変換データ、直交変換係数を、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数にそれぞれ変換する。

【0062】115はフレームレート変換制御手段であり、フレームレート変換情報より、フレームレート変換の継続／終了を判断する。そして、フレームレート変換終了後、可変長符号化手段117に、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を出力し、フレームレート変換が行われない時は、可変長復号化手段102からの量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数をそのまま出力する。

【0063】116は動きベクトル選択手段であり、上

記符号化制御・フレーム内／フレーム間判断手段113から出力された動き補償あり／なし (out) により、動きベクトルの選択を行う。すなわち、動き補償あり／なし (out) が「あり」の時は、動きベクトル (in) を後述する可変長符号化手段117にそのまま出力し、「なし」の時には可変長符号化手段117には、動きベクトル (in) を出力しない。

【0064】117は可変長符号化手段であり、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を可変長符号化することにより、ハイブリッド符号化データに変換する。また、同時に、フレーム内／フレーム間 (in) により、出力側端末の復号化装置に対して設定されたフレーム内／フレーム間識別を設定し、動き補償あり／なし (out) 、動きベクトル (in) により、出力側端末の復号化装置に対して動き補償あり／なし識別、動きベクトルを設定し、量子化係数 (in) により、出力側端末の復号化装置に対して量子化係数を設定する。

【0065】118は出力であり、ここからフレームレートの低い画像データが出力される。

【0066】ここで、Emv (i) は動き補償を伴う予測誤差直交変換データ、Q (Emv (i)) は量子化された動き補償を伴う予測誤差直交変換データ、V (Q (Emv (i))) は量子化され可変長符号化された動き補償を伴う予測誤差直交変換データ、S (i) は直交変換係数、Q (S (i)) は量子化された直交変換係数、V (Q (S (i))) は量子化され可変長符号化された直交変換係数、S' (i) は動きベクトルを考慮して演算された直交変換係数、E (i) は動き補償を伴わない予測誤差直交変換データ、Q (E (i)) は量子化された動き補償を伴わない予測誤差直交変換データ、V (Q (E (i))) は量子化され可変長符号化された動き補償を伴わない予測誤差直交変換データ、Emv\_nonmv (i) は動き補償を伴う予測誤差直交変換データ、または動き補償を伴わない予測誤差直交変換データ、Q (Emv\_nonmv (i)) は量子化された動き補償を伴う予測誤差直交変換データ、または量子化された動き補償を伴わない予測誤差直交変換データ、V (Q (Emv\_nonmv (i))) は量子化され可変長符号化された動き補償を伴う予測誤差直交変換データ、または量子化され可変長符号化された動き補償を伴う予測誤差直交変換データ、SEmv (i) は動き補償を伴う予測誤差直交変換データ、または直交変換係数、Q (SEmv (i)) は量子化された動き補償を伴う予測誤差直交変換データ、または量子化された直交変換係数、V (Q (SEmv (i))) は量子化され可変長符号化された動き補償を伴う予測誤差直交変換データ、または量子化され可変長符号化された直交変換係数、SEmv\_nonmv (i) は動き補償を伴う予測



誤差直交変換データ、または、動き補償を伴わない予測誤差直交変換データ、あるいは直交変換係数、 $Q(S_{Emv\_nonmv}(i))$ は量子化された動き補償を伴う予測誤差直交変換データ、または量子化された動き補償を伴わない予測誤差直交変換データ、あるいは量子化された直交変換係数、 $V(Q(S_{Emv\_nonmv}(i)))$ は量子化され可変長符号化された動き補償を伴う予測誤差直交変換データ、または量子化され可変長符号化された動き補償を伴わない予測誤差直交変換データ、または量子化され可変長符号化された直交変換係数とする。

【0067】フレームレート変換しない0～iフレームまでは、可変長復号化手段102、逆量子化手段103、予測誤差直交変換データ加算手段104、直交変換係数選択手段105、動き補償あり／なし(in)判断手段107、動きベクトル・直交変換係数演算手段108を経由して画像メモリ106に $S(i)$ として記憶される。また、初期画像取込選択手段109を経由し、画像メモリ110にも $S(i)$ として記憶される。また、可変長復号化手段102により可変長復号化された $Q(S_{Emv\_nonmv}(0))$ から $Q(S_{Emv\_nonmv}(i))$ は、フレームレート変換制御手段115により選択されて、可変長符号化手段117により可変長符号化されて、 $V(Q(S_{Emv\_nonmv}(0)))$ から $V(Q(S_{Emv\_nonmv}(i)))$ として出力される。

【0068】一方、フレームレート変換を行う(i+1)フレームは、可変長復号化手段102により可変長復号化され、量子化された直交変換係数 $Q(S(i+1))$ に変換される。

【0069】そして、 $Q(S(i+1))$ は逆量子化手段103により、逆量子化され、直交変換係数 $S(i+1)$ に変換される。この、 $S(i+1)$ は、直交変換係数選択手段105にて、「フレーム内」と判断され、画像メモリ106に $S(i+1)$ として記憶される。

【0070】(i+2)から(i+n)番目のフレームまでについても、上記(i+1)番目のフレームと同様の処理を繰り返すことで処理される。

【0071】そして、上記(i+n)フレームの処理が終了した時点で、画像メモリ106には直交変換係数 $S(i+n)$ が記憶されており、画像メモリ110には直交変換係数 $S(i)$ が記憶されている。

【0072】上記直交変換係数 $S(i+n)$ はフレーム内／フレーム間(in)判断手段111で「フレーム内」と判断され、 $S(i+n)$ がそのまま出力される。ここで、直交変換係数 $S(i+n)$ を新たな(i+1)番目のフレームとし、 $S(i+1)=S(i+n)$ とする。

【0073】上記直交変換係数 $S(i+1)$ は、符号化制御・フレーム内／フレーム間判断手段113にて、フ

レーム内／フレーム間(in)を用いることにより、「フレーム内」と判断され、選択される。フレームレート変換中は、動き補償あり／なし(out)は「なし」に設定される。画像取込指示は量子化手段114に $S(i+1)$ が出力された後、出力される。

【0074】 $S(i+1)$ は量子化手段114により量子化されて、量子化された直交変換係数 $Q(S(i+1))$ に変換される。

【0075】この量子化された直交変換係数 $Q(S(i+1))$ は、可変長符号化手段117により、可変長符号化され、量子化され可変長符号化された直交変換係数 $V(Q(S(i+1)))$ として出力される。

【0076】一方、フレームレート変換しない(i+n+1)番目のフレームは、可変長復号化手段102、逆量子化手段103、予測誤差直交変換データ加算手段104、直交変換係数選択手段105、動き補償あり／なし(in)判断手段107、動きベクトル・直交変換係数演算手段108を経由して画像メモリ106に直交変換係数 $S(i+n+1)$ として記憶される。また、初期画像取込選択手段109を経由して画像メモリ110にも直交変換係数 $S(i+n+1)$ として記憶される。また、可変長復号化手段102により可変長復号化されて得られた、 $Q(S_{Emv\_nonmv}(i+n+1))$ はフレームレート変換制御手段115により選択されて、 $Q(S_{Emv\_nonmv}(i+2))$ となり、さらに可変長符号化手段117により可変長符号化されて、 $V(Q(S_{Emv\_nonmv}(i+2)))$ として出力される。そして、(i+n+1)より後のフレームについても上記(i+n+1)番目のフレームと同様の処理が行われる。

【0077】以上のようにして、フレームレート変換部123で、フレームレートの高いハイブリッド符号化された画像データを、フレームレートの低いハイブリッド符号化された画像データに変換する際に、通信回線の伝送速度と端末の処理能力とを考慮し、最も律速となる処理速度に合わせてフレームレートを変換するようにしたから、通信回線の使用効率が悪化することがなくなる。

【0078】また、ハイブリッド符号化された画像データの中の任意の開始フレーム、任意の合成フレーム枚数を制御することにより、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータ、及びフレーム内符号化されたデータに対して、該データのみを選択的にフレームレート変換を行うことができ、演算量が大幅に減少し、高速で負荷の少ないフレームレート変換を行うことができる。参考までに、勧告H. 261の復号器、および符号器において全ての演算量を100%とすると、逆直交変換：約10%、直交変換：約10%、動き検出：約45%の演算量が必要であるとの報告(Interface 96年1月号)もあり、本実施の形態によれば演算量が大幅に減ることが期待できる。



【0079】実施の形態2. 以下、本発明の請求項2に対応する実施の形態2. に係るフレームレート変換方式について、図面を参照しながら説明する。

【0080】図3は入力されたフレームレートの高い画像データで、 $(i+1)$  から  $(i+n)$  番目のフレームまでが、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータである場合の、 $(i+n)$  番目のフレームの処理時のフレームレート変換方式を示すものである。

【0081】図3において、201は入力であり、フレームレートの高い画像データが入力される。

【0082】202は可変長復号化手段であり、ハイブリッド符号化データを可変長復号化することにより、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数に変換する。また、同時に、入力側の端末の符号化装置で設定されたフレーム内／フレーム間識別を検出して、フレーム内／フレーム間  $(i+n)$  を設定し、また、入力側端末の符号化装置で設定された動き補償あり／なし識別を検出して、動き補償あり／なし  $(i+n)$  を設定し、また、入力側端末の符号化装置で設定された量子化係数を検出して、量子化係数  $(i+n)$  を設定し、さらに、入力側端末の符号化装置で設定された動きベクトルを検出して、動きベクトル  $(i+n)$  を設定する。

【0083】203は逆量子化手段であり、上記設定された量子化係数  $(i+n)$  により、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を、動き補償を伴う予測誤差直交変換データ、動き補償を伴わない予測誤差直交変換データ、直交変換係数にそれぞれ変換する。

【0084】204は予測誤差直交変換データ加算手段であり、動きベクトル・直交変換係数演算手段208にて動きベクトルを考慮して演算された直交変換係数、または画像メモリ206に記憶されている直交変換係数と、動き補償を伴う予測誤差直交変換データ、または動き補償を伴わない予測誤差直交変換データとを加算する。

【0085】205は直交変換係数選択手段であり、上記設定されたフレーム内／フレーム間  $(i+n)$  により制御される。すなわち、上記フレーム内／フレーム間  $(i+n)$  が「フレーム内」の時は、逆量子化手段203により逆量子化された直交変換係数が選択され、上記フレーム内／フレーム間  $(i+n)$  が「フレーム間」の時は予測誤差直交変換データ加算手段204により算出された直交変換係数が選択され出力される。

【0086】206は画像メモリであり、上記直交変換係数選択手段205により選択された直交変換係数を記憶するためのものである。

【0087】207は動き補償あり／なし  $(i+n)$  判断手段であり、動き補償が「あり」の時は画像メモリ206に記憶されている直交変換係数を動きベクトル・直交変換係数演算手段208に出力する。動き補償が「なし」の時は画像メモリ206に記憶されている直交変換係数を予測誤差直交変換データ加算手段204に出力する。

【0088】208は動きベクトル・直交変換係数演算手段であり、動きベクトル  $(i+n)$  を用いて、動き補償あり／なし  $(i+n)$  判断手段207にて「あり」と判断された画像メモリ206に記憶されていた直交変換係数を、マトリクス演算することにより、動きベクトルを考慮して演算された直交変換係数を算出する。マトリクス演算については後述の方法を用いる。

【0089】209は初期画像取込選択手段であり、後述する符号化制御・フレーム内／フレーム間判断手段213により設定された画像取込指示により、画像メモリ206に記憶されている直交変換係数を取り込み、後述する画像メモリ210に直交変換係数を出力する。

【0090】210は画像メモリであり、初期画像取込選択手段209により選択された直交変換係数を記憶するためのものである。

【0091】211はフレーム内／フレーム間  $(i+n)$  判断手段であり、「フレーム内」の時は、画像メモリ206の直交変換係数を後段の符号化制御・フレーム内／フレーム間判断手段213へ出力し、「フレーム間」の時は、画像メモリ206の直交変換係数を後述する直交変換係数減算手段212へ出力する。

【0092】212は直交変換係数減算手段であり、フレーム内／フレーム間  $(i+n)$  判断手段211で、「フレーム間」と判断されたときに画像メモリ206に記憶されていた直交変換係数から、画像メモリ210の直交変換係数を減算する。

【0093】213は符号化制御・フレーム内／フレーム間判断手段であり、フレームレート変換が行われていない時には、動き補償あり／なし  $(i+n)$  をそのまま動き補償あり／なし  $(out)$  として出力し、フレームレート変換が行われている時には、動き補償あり／なし  $(out)$  は「なし」に設定して出力する。また、フレームレート変換を行う1つ前のフレームを画像メモリ210に記憶させるために、画像取込指示を出力する。また、フレーム内／フレーム間  $(i+n)$  を用いて、フレーム内／フレーム間  $(i+n)$  判断手段211からの直交変換係数と、直交変換係数減算手段212からの動き補償を伴わない予測誤差直交変換データとの選択を行い、後述する量子化手段214へ出力する。また、量子化係数  $(i+n)$  を再設定し、量子化係数  $(out)$  として後述する量子化手段214へ出力する。

【0094】214は量子化手段であり、上記出力された量子化係数  $(out)$  により、動き補償を伴わない予



測誤差直交変換データ、直交変換係数を、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数にそれぞれ変換する。

【0095】215はフレームレート変換制御手段であり、フレームレート変換情報より、フレームレート変換の継続／終了を判断する。そして、フレームレート変換終了後、可変長符号化手段217に、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を出力し、フレームレート変換が行われない時は、可変長復号化手段202からの量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数をそのまま出力する。

【0096】216は動きベクトル選択手段であり、上記符号化制御・フレーム内／フレーム間判断手段213から出力された動き補償あり／なし(out)により、動きベクトルの選択を行う。すなわち、動き補償あり／なし(out)が「あり」の時は、動きベクトル(in)を可変長符号化手段217にそのまま出力し、「なし」の時には、可変長符号化手段217には動きベクトル(in)を出力しない。

【0097】217は可変長符号化手段であり、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を可変長符号化することにより、ハイブリッド符号化データに変換する。また、同時に、フレーム内／フレーム間(in)により、出力側端末の復号化装置に対して設定されたフレーム内／フレーム間識別を設定し、動き補償あり／なし(out)、動きベクトル(in)により、出力側端末の復号化装置に対して、動き補償あり／なし識別、動きベクトルを設定し、量子化係数(in)により、出力側端末の復号化装置に対して量子化係数を設定する。

【0098】218は出力であり、ここからフレームレートの低い画像データが出力される。

【0099】以下、フレームレート変換処理の詳細について説明する。入力される画像データを、0からi番目のフレームまではフレームレート変換なしとし、(i+1)から(i+n)番目のフレームまでを合成して、フレームレート変換を行い、(i+n+1)番目のフレーム以降のフレームは、フレームレート変換なしとして説明する。

【0100】フレームレート変換しない0～iフレームまでは、可変長復号化手段202、逆量子化手段203、予測誤差直交変換データ加算手段204、直交変換係数選択手段205、動き補償あり／なし(in)判断手段207、動きベクトル・直交変換係数演算手段208を経由して画像メモリ206にS(i)として記憶される。また、初期画像取込選択手段209を経由し、画像メモリ210にもS(i)として記憶される。また、

可変長復号化手段202により可変長復号化されたQ(Emv(0))からQ(Emv(i))は、フレームレート変換制御手段215により選択されて、可変長符号化手段217により可変長符号化されて、V(Q(Emv(0)))からV(Q(Emv(i)))として出力される。

【0101】一方、フレームレート変換を行う(i+1)フレームは、可変長復号化手段202により可変長復号化され、量子化された動き補償を伴う予測誤差直交変換データQ(Emv(i+1))に変換される。

【0102】そして、Q(Emv(i+1))は逆量子化手段203により、逆量子化され、動き補償を伴う予測誤差直交変換データEmv(i+1)に変換される。

【0103】このEmv(i+1)は、予測誤差直交変換データ加算手段204により、動き補償あり／なし(in)判断手段207で「あり」と判断される画像メモリ206に記憶されているS(i)を、動きベクトル・直交変換係数演算手段208で処理したS'(i)と加算され、直交変換係数選択手段205にて、「フレーム間」と判断され、画像メモリ206にS(i+1)として記憶される。

【0104】(i+2)から(i+n)番目のフレームまでについても、上記(i+1)番目のフレームと同様の処理を繰り返すことで処理される。

【0105】そして、上記(i+n)フレームの処理が終了した時点で、画像メモリ206には直交変換係数S(i+n)が記憶されており、画像メモリ210には直交変換係数S(i)が記憶されている。

【0106】上記直交変換係数S(i+n)は、フレーム内／フレーム間(in)判断手段211で、「フレーム間」と判断され、直交変換係数減算手段212により、S(i+n)-S(i)の処理が行われ、S(n)が算出される。ここで、S(n)は、動き補償を伴わない予測誤差直交変換データであり、(i+1)番目のフレームのデータとすると、E(i+1)=S(n)とすることができる。

【0107】上記動き補償を伴わない予測誤差直交変換データE(i+1)は符号化制御・フレーム内／フレーム間判断手段213にて、フレーム内／フレーム間(in)を用いることにより、「フレーム間」と判断され、選択される。フレームレート変換中は、動き補償あり／なし(out)は「なし」に設定される。画像取込指示は量子化手段214にE(i+1)が出力された後、出力される。

【0108】E(i+1)は量子化手段214により量子化されて、量子化された動き補償を伴わない予測誤差直交変換データQ(E(i+1))に変換される。

【0109】この量子化された動き補償を伴わない予測誤差直交変換データQ(E(i+1))は可変長符号化手段217により、可変長符号化され、量子化され可変



長符号化された動き補償を伴わない予測誤差直交変換データ  $V(Q(E(i+1)))$  として出力される。

【0110】一方、フレームレート変換しない  $(i+n+1)$  番目のフレームは、可変長復号化手段 202、逆量子化手段 203、予測誤差直交変換データ加算手段 204、直交変換係数選択手段 205、動き補償あり／なし  $(i_n)$  判断手段 207、動きベクトル・直交変換係数演算手段 208 を経由して、画像メモリ 206 に直交変換係数  $S(i+n+1)$  として記憶される。また、初期画像取込選択手段 209 を経由して画像メモリ 210 にも  $S(i+n+1)$  として記憶される。また、可変長復号化手段 202 により可変長復号化されて得られた  $Q(E_{mv}(i+n+1))$  は、フレームレート変換制御手段 215 により選択されて、 $Q(E_{mv}(i+2))$  となり、さらに可変長符号化手段 217 により可変長符号化されて、 $V(Q(E_{mv}(i+2)))$  として出力される。そして、 $(i+n+1)$  より後のフレームについても上記  $(i+n+1)$  番目のフレームと同様の処理が行われる。

【0111】以上のように、本実施の形態によれば、ハイブリッド符号化された画像データの中の任意の開始フレーム、任意の合成フレーム枚数を制御することにより、動き補償を伴うフレーム間予測誤差に対して、該データのみを選択的にフレームレート変換を行うことができ、演算量が大幅に減少し、高速で負荷の少ないフレームレート変換を行うことができる。

【0112】実施の形態 3. 以下、本発明の請求項 3 に対応する実施の形態 3. に係るフレームレート変換方式について、図面を参照しながら説明する。

【0113】図 4 は入力されたフレームレートの高い画像データで、 $(i+1)$  から  $(i+n)$  番目のフレームまでが、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータとの両方のデータが含まれ、 $(i+1)$ 、 $(i+n)$  番目のフレームが、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータである場合の  $(i+n)$  番目のフレームの処理時のフレームレート変換方式を示すものである。

【0114】図 4 において、301 は入力であり、フレームレートの高い画像データが入力される。

【0115】302 は可変長復号化手段であり、ハイブリッド符号化データを可変長復号化することにより、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数に変換する。また、同時に、入力側端末の符号化装置で設定されたフレーム内／フレーム間識別を検出して、フレーム内／フレーム間  $(i_n)$  を設定し、また、入力側端末の符号化装置で設

定された動き補償あり／なし識別を検出して、動き補償あり／なし  $(i_n)$  を設定し、また、入力側端末の符号化装置で設定された量子化係数を検出して、量子化係数  $(i_n)$  を設定し、さらに、入力側端末の符号化装置で設定された動きベクトルを検出して、動きベクトル  $(i_n)$  を設定する。

【0116】303 は逆量子化手段であり、上記設定された量子化係数  $(i_n)$  により、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を、動き補償を伴う予測誤差直交変換データ、動き補償を伴わない予測誤差直交変換データ、直交変換係数にそれぞれ変換する。

【0117】304 は予測誤差直交変換データ加算手段であり、動きベクトル・直交変換係数演算手段 308 にて動きベクトルを考慮して演算された直交変換係数、または画像メモリ 306 に記憶されている直交変換係数と、動き補償を伴う予測誤差直交変換データ、または動き補償を伴わない予測誤差直交変換データとを加算する。

【0118】305 は直交変換係数選択手段であり、上記設定されたフレーム内／フレーム間  $(i_n)$  により制御される。すなわち、上記フレーム内／フレーム間  $(i_n)$  が「フレーム内」の時は、逆量子化手段 303 により逆量子化された直交変換係数が選択され、上記フレーム内／フレーム間  $(i_n)$  が「フレーム間」の時は、予測誤差直交変換データ加算手段 304 により算出された直交変換係数が選択され出力される。

【0119】306 は画像メモリであり、上記直交変換係数選択手段 305 により選択された直交変換係数を記憶するためのものである。

【0120】307 は動き補償あり／なし  $(i_n)$  判断手段であり、動き補償が「あり」の時は画像メモリ 306 に記憶されている直交変換係数を動きベクトル・直交変換係数演算手段 308 に出力する。動き補償が「なし」の時は、画像メモリ 306 に記憶されている直交変換係数を予測誤差直交変換データ加算手段 304 に出力する。

【0121】308 は動きベクトル・直交変換係数演算手段であり、動きベクトル  $(i_n)$  を用いて、動き補償あり／なし  $(i_n)$  判断手段 307 にて「あり」と判断された画像メモリ 306 に記憶されていた直交変換係数を、マトリクス演算することにより、動きベクトルを考慮して演算された直交変換係数を算出する。マトリクス演算については後述の方法を用いる。

【0122】309 は初期画像取込選択手段であり、後述する符号化制御・フレーム内／フレーム間判断手段 313 により設定された画像取込指示により、画像メモリ 306 に記憶されている直交変換係数を取り込み、後述する画像メモリ 310 に直交変換係数を出力する。



【0123】310は画像メモリであり、初期画像取込選択手段309により選択された直交変換係数を記憶するためのものである。

【0124】311はフレーム内／フレーム間 (in) 判断手段であり、「フレーム内」の時は、画像メモリ306の直交変換係数を後段の符号化制御・フレーム内／フレーム間判断手段313へ出力し、「フレーム間」の時は、画像メモリ306の直交変換係数を後述する直交変換係数減算手段312へ出力する。

【0125】312は直交変換係数減算手段であり、フレーム内／フレーム間 (in) 判断手段311で、「フレーム間」と判断されたときに画像メモリ306に記憶されていた直交変換係数から、画像メモリ310の直交変換係数を減算する。

【0126】313は符号化制御・フレーム内／フレーム間判断手段であり、フレームレート変換が行われていない時には、動き補償あり／なし (in) をそのまま動き補償あり／なし (out) として出力し、フレームレート変換が行われている時には、動き補償あり／なし (out) は「なし」に設定して出力する。また、フレームレート変換を行う1つ前のフレームを画像メモリ310に記憶させるために、画像取込指示を出力する。また、フレーム内／フレーム間 (in) を用いて、フレーム内／フレーム間 (in) 判断手段311からの直交変換係数と、直交変換係数減算手段312からの動き補償を伴わない予測誤差直交変換データとの選択を行い、後述する量子化手段314へ出力する。また、量子化係数 (in) を再設定し、量子化係数 (out) として後述する量子化手段314へ出力する。

【0127】314は量子化手段であり、上記出力された量子化係数 (out) により、動き補償を伴わない予測誤差直交変換データ、直交変換係数を、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数にそれぞれ変換する。

【0128】315はフレームレート変換制御手段であり、フレームレート変換情報よりフレームレート変換の継続／終了を判断する。そして、フレームレート変換終了後、可変長符号化手段317に、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を出力し、フレームレート変換が行われない時は、可変長復号化手段302からの量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数をそのまま出力する。

【0129】316は動きベクトル選択手段であり、上記符号化制御・フレーム内／フレーム間判断手段313から出力された動き補償あり／なし (out) により、動きベクトルの選択を行う。すなわち、動き補償あり／なし (out) が「あり」の時は、動きベクトル (in) を可変長符号化手段317にそのまま出力し、「な

し」の時には、可変長符号化手段317には動きベクトル (in) を出力しない。

【0130】317は可変長符号化手段であり、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を可変長符号化することにより、ハイブリッド符号化データに変換する。また、同時に、フレーム内／フレーム間 (in) により、出力側端末の復号化装置に対して設定されたフレーム内／フレーム間識別を設定し、動き補償あり／なし (out) 、動きベクトル (in) により、出力側端末の復号化装置に対して動き補償あり／なし識別、動きベクトルを設定し、量子化係数 (in) により、出力側端末の復号化装置に対して量子化係数を設定する。

【0131】318は出力であり、ここからフレームレートの低い画像データが出力される。

【0132】以下、フレームレート変換処理の詳細について説明する。入力される画像データを0からi番目のフレームまではフレームレート変換なしとし、(i+1) から (i+n) 番目のフレームまでを合成して、フレームレート変換を行い、(i+n+1) 番目のフレーム以降のフレームは、フレームレート変換なしとして説明する。

【0133】フレームレート変換しない0～iフレームまでは、可変長復号化手段302、逆量子化手段303、予測誤差直交変換データ加算手段304、直交変換係数選択手段305、動き補償あり／なし (in) 判断手段307、動きベクトル・直交変換係数演算手段308を経由して画像メモリ306にS(i)として記憶される。また、初期画像取込選択手段309を経由し、画像メモリ310にもS(i)として記憶される。また、可変長復号化手段302により可変長復号化されたQ(Emv\_\_nonmv(0))からQ(Emv\_\_nonmv(i))は、フレームレート変換制御手段315により選択されて、可変長符号化手段317により可変長符号化されて、V(Q(Emv\_\_nonmv(0)))からV(Q(Emv\_\_nonmv(i)))として出力される。

【0134】一方、フレームレート変換を行う(i+1)フレームは、可変長復号化手段302により、可変長復号化され、量子化された動き補償を伴わない予測誤差直交変換データQ(E(i+1))に変換される。

【0135】そして、Q(E(i+1))は逆量子化手段303により、逆量子化され、E(i+1)に変換される。

【0136】このE(i+1)は、予測誤差直交変換データ加算手段304により、動き補償あり／なし (in) 判断手段307で「あり」と判断される画像メモリ306に記憶されているS(i)を、動きベクトル・直交変換係数演算手段308で処理したS'(i)、また



は動き補償あり／なし (i n) 判断手段307で「なし」と判断される画像メモリ306に記憶されているS (i) と加算され、直交変換係数選択手段305にて、「フレーム間」と判断され、画像メモリ306にS (i + 1) として記憶される。

【0137】 (i + 2) から (i + n) 番目のフレームまでについては動き補償を伴わない場合は上記 (i + 1) 番目のフレーム、動き補償を伴う場合は上記実施の形態2. と同様の処理を繰り返すことで処理される。

【0138】そして、上記 (i + n) フレームの処理が終了した時点で、画像メモリ306には直交変換係数S (i + n) が記憶されており、画像メモリ310には直交変換係数S (i) が記憶されている。

【0139】上記直交変換係数S (i + n) は、フレーム内／フレーム間 (i n) 判断手段311で「フレーム間」と判断され、直交変換係数減算手段312により、 $S(i+n) - S(i)$  の処理が行われ、S (n) が算出される。ここで、S (n) は、動き補償を伴わない予測誤差直交変換データであり、(i + 1) 番目のフレームのデータとすると、 $E(i+1) = S(n)$  とすることができる。

【0140】上記動き補償を伴わない予測誤差直交変換データE (i + 1) は符号化制御・フレーム内／フレーム間判断手段313にて、フレーム内／フレーム間 (i n) を用いることにより、「フレーム間」と判断され、選択される。フレームレート変換中は、動き補償あり／なし (out) は「なし」に設定される。画像取込指示は量子化手段314にE (i + 1) が出力された後、出力される。

【0141】E (i + 1) は量子化手段314により、量子化されて、量子化された動き補償を伴わない予測誤差直交変換データQ (E (i + 1)) に変換される。

【0142】この量子化された動き補償を伴わない予測誤差直交変換データQ (E (i + 1)) は可変長符号化手段317により、可変長符号化され、量子化され可変長符号化された動き補償を伴わない予測誤差直交変換データV (Q (E (i + 1))) として出力される。

【0143】一方、フレームレート変換しない (i + n + 1) 番目のフレームは、可変長復号化手段302、逆量子化手段303、予測誤差直交変換データ加算手段304、直交変換係数選択手段305、動き補償あり／なし (i n) 判断手段307、動きベクトル・直交変換係数演算手段308を経由して画像メモリ306に直交変換係数S (i + n + 1) として記憶される。また、初期画像取込選択手段309を経由して画像メモリ310にもS (i + n + 1) として記憶される。また、可変長復号化手段302により可変長復号化されて得られたQ (Emv\_\_nonmv (i + n + 1)) は、フレームレート変換制御手段315により選択されて、Q (Emv\_\_nonmv (i + 2)) となり、さらに可変長符号化

手段317により可変長符号化されて、V (Q (Emv\_\_nonmv (i + 2))) として出力される。そして、(i + n + 1) より後のフレームについても上記 (i + n + 1) 番目のフレームと同様の処理が行われる。

【0144】以上のように、本実施の形態によれば、ハイブリッド符号化された画像データの中の任意の開始フレーム、任意の合成フレーム枚数を制御することにより、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータ、及び動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータの両方のデータが含まれるような場合においても、該データのみを選択的にフレームレート変換を行うことができ、演算量が大幅に減少し、高速で負荷の少ないフレームレート変換を行うことができる。

【0145】実施の形態4. 以下、本発明の請求項4に対応する実施の形態4. に係るフレームレート変換方式について、図面を参照しながら説明する。

【0146】図5は入力されたフレームレートの高い画像データで (i + 1) から (i + n) 番目のフレームまでが、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータと、フレーム内符号化されたデータとの両方のデータが含まれ、(i + n) 番目のフレームがフレーム内符号化されたデータである場合の (i + n) 番目のフレームの処理時のフレームレート変換方式を示すものである。

【0147】図5において、401は入力であり、フレームレートの高い画像データが入力される。

【0148】402は可変長復号化手段であり、ハイブリッド符号化データを可変長復号化することにより、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数に変換する。また、同時に、入力側端末の符号化装置で設定されたフレーム内／フレーム間識別を検出して、フレーム内／フレーム間 (i n) を設定し、また、入力側端末の符号化装置で設定された動き補償あり／なし識別を検出して、動き補償あり／なし (i n) を設定し、また、入力側端末の符号化装置で設定された量子化係数を検出して、量子化係数 (i n) を設定し、さらに、入力側端末の符号化装置で設定された動きベクトルを検出して、動きベクトル (i n) を設定する。

【0149】403は逆量子化手段であり、上記設定された量子化係数 (i n) により、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を、動き補償を伴う予測誤差直交変換データ、動き補償を伴わない予測誤差直交変換データ、直交変換係数にそれぞれ変換する。



【0150】404は予測誤差直交変換データ加算手段であり、動きベクトル・直交変換係数演算手段408にて動きベクトルを考慮して演算された直交変換係数、または画像メモリ406に記憶されている直交変換係数と、動き補償を伴う予測誤差直交変換データ、または動き補償を伴わない予測誤差直交変換データとを加算する。

【0151】405は直交変換係数選択手段であり、上記設定されたフレーム内／フレーム間 (in) により制御される。すなわち、上記フレーム内／フレーム間 (in) が「フレーム内」の時は、逆量子化手段403により逆量子化された直交変換係数が選択され、上記フレーム内／フレーム間 (in) が「フレーム間」の時は、予測誤差直交変換データ加算手段404により算出された直交変換係数が選択され出力される。

【0152】406は画像メモリであり、上記直交変換係数選択手段405により選択された直交変換係数を記憶するためのものである。

【0153】407は動き補償あり／なし (in) 判断手段であり、動き補償が「あり」の時は画像メモリ406に記憶されている直交変換係数を動きベクトル・直交変換係数演算手段408に出力する。動き補償が「なし」の時は画像メモリ406に記憶されている直交変換係数を予測誤差直交変換データ加算手段404に出力する。

【0154】408は動きベクトル・直交変換係数演算手段であり、動きベクトル (in) を用いて、動き補償あり／なし (in) 判断手段407にて「あり」と判断された画像メモリ406に記憶されていた直交変換係数を、マトリクス演算することにより、動きベクトルを考慮して演算された直交変換係数を算出する。マトリクス演算については後述の方法を用いる。

【0155】409は初期画像取込選択手段であり、後述する符号化制御・フレーム内／フレーム間判断手段413により設定された画像取込指示により、画像メモリ406に記憶されている直交変換係数を取り込み、後述する画像メモリ410に直交変換係数を出力する。

【0156】410は画像メモリであり、初期画像取込選択手段409により選択された直交変換係数を記憶するためのものである。

【0157】411はフレーム内／フレーム間 (in) 判断手段であり、「フレーム内」の時は、画像メモリ406の直交変換係数を後段の符号化制御・フレーム内／フレーム間判断手段413へ出力し、「フレーム間」の時は、画像メモリ406の直交変換係数を後述する直交変換係数減算手段412へ出力する。

【0158】412は直交変換係数減算手段であり、フレーム内／フレーム間 (in) 判断手段411で、「フレーム間」と判断されたときに画像メモリ406に記憶されていた直交変換係数から、画像メモリ410の直交

変換係数を減算する。

【0159】413は符号化制御・フレーム内／フレーム間判断手段であり、フレームレート変換が行われていない時には、動き補償あり／なし (in) をそのまま動き補償あり／なし (out) として出力し、フレームレート変換が行われている時には、動き補償あり／なし (out) は「なし」に設定して出力する。また、フレームレート変換を行う1つ前のフレームを画像メモリ410に記憶させるために、画像取込指示を出力する。また、フレーム内／フレーム間 (in) を用いて、フレーム内／フレーム間 (in) 判断手段411からの直交変換係数と、直交変換係数減算手段412からの動き補償を伴わない予測誤差直交変換データとの選択を行い、量子化手段414へ出力する。また、量子化係数 (in) を再設定し、量子化係数 (out) として後述する量子化手段414へ出力する。

【0160】414は量子化手段であり、上記出力された量子化係数 (out) により、動き補償を伴わない予測誤差直交変換データ、直交変換係数を、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数にそれぞれ変換する。

【0161】415はフレームレート変換制御手段であり、フレームレート変換情報よりフレームレート変換の継続／終了を判断する。そして、フレームレート変換終了後、可変長符号化手段417に、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を出力し、フレームレート変換が行われていない時は、可変長復号化手段402からの量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数をそのまま出力する。

【0162】416は動きベクトル選択手段であり、上記符号化制御・フレーム内／フレーム間判断手段413から出力された動き補償あり／なし (out) により、動きベクトルの選択を行う。すなわち、動き補償あり／なし (out) が「あり」の時は、動きベクトル (in) を可変長符号化手段417にそのまま出力し、「なし」の時には、可変長符号化手段417には動きベクトル (in) を出力しない。

【0163】417は可変長符号化手段であり、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を可変長符号化することにより、ハイブリッド符号化データに変換する。また、同時に、フレーム内／フレーム間 (in) により、出力側端末の復号化装置に対して設定されたフレーム内／フレーム間識別を設定し、動き補償あり／なし (out)、動きベクトル (in) により、出力側端末の復号化装置に対して動き補償あり／なし識別、動きベクトルを設定し、量子化係数 (in) により、出力側端末の復号化装



置に対して量子化係数を設定する。

【0164】418は出力であり、ここからフレームレートの低い画像データが出力される。

【0165】以下、フレームレート変換処理の詳細について説明する。入力される画像データを0から*i*番目のフレームまではフレームレート変換なしとし、(*i*+1)から(*i*+*n*)番目のフレームまでを合成して、フレームレート変換を行い、(*i*+*n*+1)番目のフレーム以降のフレームは、フレームレート変換なしとして説明する。

【0166】フレームレート変換しない0~*i*フレームまでは、可変長復号化手段402、逆量子化手段403、予測誤差直交変換データ加算手段404、直交変換係数選択手段405、動き補償あり／なし(*i**n*)判断手段407、動きベクトル・直交変換係数演算手段408を経由して画像メモリ406に*S*(*i*)として記憶される。また、初期画像取込選択手段409を経由し、画像メモリ410にも*S*(*i*)として記憶される。また、可変長復号化手段402により可変長復号化された*Q*(*S*Emv(0))から*Q*(*S*Emv(*i*))は、フレームレート変換制御手段415により選択されて、可変長符号化手段417により可変長符号化されて、*V*(*Q*(*S*Emv(0)))から*V*(*Q*(*S*Emv(*i*))))として出力される。

【0167】一方、フレームレート変換を行う(*i*+1)フレームは、可変長復号化手段402により、可変長復号化され、量子化された直交変換係数*Q*(*S*(*i*+1)))に変換される。

【0168】そして、*Q*(*S*(*i*+1)))は逆量子化手段403により、逆量子化され、*S*(*i*+1))に変換される。

【0169】この*S*(*i*+1))は、直交変換係数選択手段405にて、「フレーム内」と判断され、画像メモリ406に*S*(*i*+1))として記憶される。

【0170】(*i*+2)から(*i*+*n*)番目のフレームまでについてはフレーム内の場合には上記(*i*+1)番目のフレーム、動き補償を伴う場合は上記実施の形態2と同様の処理を繰り返すことで処理される。

【0171】そして、上記(*i*+*n*)フレームの処理が終了した時点で、画像メモリ406には直交変換係数*S*(*i*+*n*)が記憶されており、画像メモリ410には直交変換係数*S*(*i*)が記憶されている。

【0172】上記直交変換係数*S*(*i*+*n*)は、フレーム内／フレーム間(*i**n*)判断手段411で「フレーム内」と判断され、*S*(*i*+*n*)が出力される。ここで、*S*(*i*+*n*)を新たな(*i*+1)番目のフレームとし、*S*(*i*+1)=*S*(*i*+*n*)とする。

【0173】上記直交変換係数*S*(*i*+1))は、符号化制御・フレーム内／フレーム間判断手段413にて、フレーム内／フレーム間(*i**n*)を用いることにより、

「フレーム内」と判断され、選択される。フレームレート変換中は、動き補償あり／なし(out)は「なし」に設定される。画像取込指示は量子化手段414に*S*(*i*+1))が出力された後、出力される。

【0174】*S*(*i*+1))は量子化手段414により、量子化されて、量子化された直交変換係数*Q*(*S*(*i*+1)))に変換される。

【0175】この量子化された上記直交変換係数*Q*(*S*(*i*+1)))は可変長符号化手段417により、可変長符号化され、*V*(*Q*(*S*(*i*+1))))として出力される。

【0176】一方、フレームレート変換しない(*i*+*n*+1)番目のフレームは、可変長復号化手段402、逆量子化手段403、予測誤差直交変換データ加算手段404、直交変換係数選択手段405、動き補償あり／なし(*i**n*)判断手段407、動きベクトル・直交変換係数演算手段408を経由して画像メモリ406に直交変換係数*S*(*i*+*n*+1))として記憶される。また、初期画像取込選択手段409を経由して画像メモリ410にも*S*(*i*+*n*+1))として記憶される。また、可変長復号化手段402により可変長復号化されて得られた*Q*(*S*Emv(*i*+*n*+1)))は、フレームレート変換制御手段415により選択されて、*Q*(*S*Emv(*i*+2)))となり、さらに可変長符号化手段417により可変長符号化されて、*V*(*Q*(*S*Emv(*i*+2))))として出力される。そして、(*i*+*n*+1)より後のフレームについても上記(*i*+*n*+1)番目のフレームと同様の処理が行われる。

【0177】以上のように、本実施の形態によれば、ハイブリッド符号化された画像データの中の任意の開始フレーム、任意の合成フレーム枚数を制御することにより、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータ、及びフレーム内符号化されたデータの両方のデータが含まれる場合においても、該データのみを選択的にフレームレート変換を行うことができ、演算量が大幅に減少し、高速で負荷の少ないフレームレート変換を行うことができる。

【0178】実施の形態5. 以下、本発明の請求項5に対応する実施の形態5.に係るフレームレート変換方式について、図面を参照しながら説明する。

【0179】図6は入力されたフレームレートの高い画像データで(*i*+1)から(*i*+*n*)番目のフレームまでが、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータと、さらには、フレーム内符号化されたデータの3種類のデータが含まれ、(*i*+1)番目のフレームがフレーム内符号化されたデータであり、(*i*+*n*-1)番目のフレームが動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイ



ブリッド符号化されたデータフレーム内符号化されたデータであり、 $(i+n)$  番目のフレームが動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータである場合の  $(i+n)$  番目のフレームの処理時のフレームレート変換方式を示すものである。

【0180】図6において、501は入力であり、フレームレートの高い画像データが入力される。

【0181】502は可変長復号化手段であり、ハイブリッド符号化データを可変長復号化することにより、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数に変換する。また、同時に、入力側端末の符号化装置で設定されたフレーム内／フレーム間識別を検出して、フレーム内／フレーム間  $(i+n)$  を設定し、また、入力側端末の符号化装置で設定された動き補償あり／なし識別を検出して、動き補償あり／なし  $(i+n)$  を設定し、また、入力側端末の符号化装置で設定された量子化係数を検出して、量子化係数  $(i+n)$  を設定し、さらに、入力側端末の符号化装置で設定された動きベクトルを検出して、動きベクトル  $(i+n)$  を設定する。

【0182】503は逆量子化手段であり、上記設定された量子化係数  $(i+n)$  により、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を、動き補償を伴う予測誤差直交変換データ、動き補償を伴わない予測誤差直交変換データ、直交変換係数にそれぞれ変換する。

【0183】504は予測誤差直交変換データ加算手段であり、動きベクトル・直交変換係数演算手段508にて動きベクトルを考慮して演算された直交変換係数、または画像メモリ506に記憶されている直交変換係数と、動き補償を伴う予測誤差直交変換データ、または動き補償を伴わない予測誤差直交変換データとを加算する。

【0184】505は直交変換係数選択手段であり、上記設定されたフレーム内／フレーム間  $(i+n)$  により制御される。すなわち、上記フレーム内／フレーム間  $(i+n)$  が「フレーム内」の時は、逆量子化手段503により逆量子化された直交変換係数が選択され、上記フレーム内／フレーム間  $(i+n)$  が「フレーム間」の時は、予測誤差直交変換データ加算手段504により算出された直交変換係数が選択され出力される。

【0185】506は画像メモリであり、上記直交変換係数選択手段505により選択された直交変換係数を記憶するためのものである。

【0186】507は動き補償あり／なし  $(i+n)$  判断手段であり、動き補償が「あり」の時は画像メモリ506に記憶されている直交変換係数を動きベクトル・直交

変換係数演算手段508に出力する。動き補償が「なし」の時は画像メモリ506に記憶されている直交変換係数を予測誤差直交変換データ加算手段504に出力する。

【0187】508は動きベクトル・直交変換係数演算手段であり、動きベクトル  $(i+n)$  を用いて、動き補償あり／なし  $(i+n)$  判断手段507にて「あり」と判断された画像メモリ506に記憶されていた直交変換係数を、マトリクス演算することにより、動きベクトルを考慮して演算された直交変換係数を算出する。マトリクス演算については後述の方法を用いる。

【0188】509は初期画像取込選択手段であり、後述する符号化制御・フレーム内／フレーム間判断手段513により設定された画像取込指示により、画像メモリ506に記憶されている直交変換係数を取り込み、後述する画像メモリ510に直交変換係数を出力する。

【0189】510は画像メモリであり、初期画像取込選択手段509により選択された直交変換係数を記憶するためのものである。

【0190】511はフレーム内／フレーム間  $(i+n)$  判断手段であり、「フレーム内」の時は、画像メモリ506の直交変換係数を後段の符号化制御・フレーム内／フレーム間判断手段513へ出力し、「フレーム間」の時は、画像メモリ506の直交変換係数を後述する直交変換係数減算手段512へ出力する。

【0191】512は直交変換係数減算手段であり、フレーム内／フレーム間  $(i+n)$  判断手段511で、「フレーム間」と判断されたときに画像メモリ506に記憶されていた直交変換係数から、画像メモリ510の直交変換係数を減算する。

【0192】513は符号化制御・フレーム内／フレーム間判断手段であり、フレームレート変換が行われていない時には、動き補償あり／なし  $(i+n)$  をそのまま動き補償あり／なし  $(out)$  として出力し、フレームレート変換が行われている時には、動き補償あり／なし  $(out)$  は「なし」に設定して出力する。また、フレームレート変換を行う1つ前のフレームを画像メモリ510に記憶させるために、画像取込指示を出力する。また、フレーム内／フレーム間  $(i+n)$  を用いて、フレーム内／フレーム間  $(i+n)$  判断手段511からの直交変換係数と、直交変換係数減算手段512からの動き補償を伴わない予測誤差直交変換データとの選択を行い、量子化手段514へ出力する。また、量子化係数  $(i+n)$  を再設定し、量子化係数  $(out)$  として後述する量子化手段514へ出力する。

【0193】514は量子化手段であり、上記出力された量子化係数  $(out)$  により、動き補償を伴わない予測誤差直交変換データ、直交変換係数を、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数にそれぞれ変換する。



【0194】515はフレームレート変換制御手段であり、フレームレート変換情報よりフレームレート変換の継続／終了を判断する。そして、フレームレート変換終了後、可変長符号化手段517に、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を出力し、フレームレート変換が行われない時は、可変長復号化手段502からの量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数をそのまま出力する。

【0195】516は動きベクトル選択手段であり、上記符号化制御・フレーム内／フレーム間判断手段513から出力された動き補償あり／なし(out)により、動きベクトルの選択を行う。すなわち、動き補償あり／なし(out)が「あり」の時は、動きベクトル(in)を可変長符号化手段517にそのまま出力し、「なし」の時には、可変長符号化手段517には動きベクトル(in)を出力しない。

【0196】517は可変長符号化手段であり、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を可変長符号化することにより、ハイブリッド符号化データに変換する。また、同時に、フレーム内／フレーム間(in)により、出力側端末の復号化装置に対して設定されたフレーム内／フレーム間識別を設定し、動き補償あり／なし(out)、動きベクトル(in)により、出力側端末の復号化装置に対して動き補償あり／なし識別、動きベクトルを設定し、量子化係数(in)により、出力側端末の復号化装置に対して量子化係数を設定する。

【0197】518は出力であり、ここからフレームレートの低い画像データが出力される。

【0198】以下、フレームレート変換処理の詳細について説明する。入力される画像データを0からi番目のフレームまではフレームレート変換なしとし、(i+1)から(i+n)番目のフレームまでを合成して、フレームレート変換を行い、(i+n+1)番目のフレーム以降のフレームは、フレームレート変換なしとして説明する。

【0199】フレームレート変換しない0～iフレームまでは、可変長復号化手段502、逆量子化手段503、予測誤差直交変換データ加算手段504、直交変換係数選択手段505、動き補償あり／なし(in)判断手段507、動きベクトル・直交変換係数演算手段508を経由して画像メモリ506にS(i)として記憶される。また、初期画像取込選択手段509を経由し、画像メモリ510にもS(i)として記憶される。また、可変長復号化手段502により可変長復号化されたQ(SEmv\_\_nonmv(0))からQ(SEmv\_\_nonmv(i))は、フレームレート変換制御手段51

5により選択されて、可変長符号化手段517により可変長符号化されて、V(Q(SEmv\_\_nonmv(0)))からV(Q(SEmv\_\_nonmv(i)))として出力される。

【0200】一方、フレームレート変換を行う(i+1)フレームは、可変長復号化手段502により、可変長復号化され、量子化された直交変換係数Q(S(i+1))に変換される。

【0201】そして、Q(S(i+1))は逆量子化手段503により、逆量子化され、S(i+1)に変換される。

【0202】このS(i+1)は、直交変換係数選択手段505にて、「フレーム内」と判断され、画像メモリ506にS(i+1)として記憶される。

【0203】(i+2)から(i+n)番目のフレームまでについてはフレーム内の場合には上記(i+1)番目のフレーム、動き補償を伴う場合、または動き補償を伴わない場合には上記実施の形態3.と同様の処理を繰り返すことで処理される。

【0204】そして、上記(i+n)フレームの処理が終了した時点で、画像メモリ506には直交変換係数S(i+n)が記憶されており、画像メモリ510には直交変換係数S(i)が記憶されている。

【0205】上記直交変換係数S(i+n)はフレーム内／フレーム間(in)判断手段511で「フレーム間」と判断され、直交変換係数減算手段512により、S(i+n)-S(i)の処理が行われ、S(n)が算出される。ここで、S(n)は、動き補償を伴わない予測誤差直交変換データであり、(i+1)番目のフレームのデータとすると、E(i+1)=S(n)とすることができる。

【0206】上記動き補償を伴わない予測誤差直交変換データE(i+1)は符号化制御・フレーム内／フレーム間判断手段513にて、フレーム内／フレーム間(in)を用いることにより、「フレーム間」と判断され、選択される。フレームレート変換中は、動き補償あり／なし(out)は「なし」に設定される。画像取込指示は量子化手段514にE(i+1)が出力された後、出力される。

【0207】E(i+1)は量子化手段514により、量子化されて、量子化された動き補償を伴わない予測誤差直交変換データQ(E(i+1))に変換される。

【0208】この量子化された動き補償を伴わない予測誤差直交変換データQ(E(i+1))は可変長符号化手段517により、可変長符号化され、V(Q(E(i+1)))として出力される。

【0209】一方、フレームレート変換しない(i+n+1)番目のフレームは、可変長復号化手段502、逆量子化手段503、予測誤差直交変換データ加算手段504、直交変換係数選択手段505、動き補償あり／な



し ( i n ) 判断手段 507、動きベクトル・直交変換係数演算手段 508 を経由して画像メモリ 506 に直交変換係数  $S(i+n+1)$  として記憶される。また、初期画像取込選択手段 509 を経由して、画像メモリ 510 にも  $S(i+n+1)$  として記憶される。また、可変長復号化手段 502 により可変長復号化されて得られた  $Q(S_{Emv\_nonmv}(i+n+1))$  は、フレームレート変換制御手段 515 により選択されて、 $Q(S_{Emv\_nonmv}(i+2))$  となり、さらに可変長符号化手段 517 により可変長符号化されて、 $V(Q(S_{Emv\_nonmv}(i+2)))$  として出力される。そして、 $(i+n+1)$  より後のフレームについても上記  $(i+n+1)$  番目のフレームと同様の処理が行われる。

【0210】以上のように、本実施の形態によれば、ハイブリッド符号化された画像データの中の任意の開始フレーム、任意の合成フレーム枚数を制御することにより、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータと、フレーム内符号化されたデータの 3 種類のデータが含まれる場合においても、該データのみを選択的にフレームレート変換を行うことができ、演算量が大幅に減少し、高速で負荷の少ないフレームレート変換を行うことができる。

【0211】実施の形態 6. 以下、本発明の請求項 6 に対応する実施の形態 6. に係るフレームレート変換方式について、フレームレート変換方式の構成を図面を参照しながら説明する。

【0212】図 7 の構成図は上記請求項 1 におけるフレームレート変換方式の構成を示すものである。

【0213】図 7 において、621 は端末であり、その処理能力は  $c_1$  とする。622 は上記端末 621 の後段に接続された通信回線であり、その伝送速度は  $b_1$  とする。623 は上記通信回線 622 の後段に接続されたフレームレート変換部であり、フレームレート変換情報が端末 625 から入力され、通信回線 622 の伝送速度  $b_1$  と、通信回線 624 の伝送速度  $b_2$  と端末 625 の処理能力  $c_2$  とをそれぞれ考慮したフレームレート変換を行うように構成されている。624 は通信回線であり、その伝送速度は  $b_2$  とする。また、625 は端末であり、その処理能力は  $c_2$  とする。

【0214】以下、フレームレートの高い画像データを端末 621 側から入力し、通信回線の伝送速度が、 $b_1 \geq b_2$  となる関係で、かつ、端末の処理能力が、 $c_1 = c_2$  の関係となるときのフレームレート変換方式について図 7 を参照しながら説明する。

【0215】ここで、 $c_1 = c_2$  であるので、端末同士の処理能力は等しく、これに合わせたフレームレートの変換は考慮する必要はない。よって、通信回線間の伝送

速度  $b_1$  と  $b_2$  の関係に着目することになる。

【0216】今、通信回線 624 の伝送速度  $b_2$  は通信回線 622 の伝送速度  $b_1$  以下であるので、フレームレート変換部 623 では通信回線 624 の伝送速度  $b_2$  で伝送可能なフレームレートに変換すればよいことになる。よって、端末 621 側から入力されたフレームレートの高いハイブリッド符号化された画像データは、伝送速度  $b_2$  の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末 625 側に出力される。端末 625 では、通信回線 624 の伝送速度  $b_2$  と等しいフレームレートの画像データが送られて、端末 625 の処理能力範囲で完全に再生されて表示されることになる。

【0217】次に、フレームレートの高い画像データを端末 621 側から入力し、通信回線の伝送速度が、 $b_1 \geq b_2$  となる関係で、かつ、端末の処理能力が、 $c_1 > c_2$  の関係となるときのフレームレート変換方式について図 7 を参照しながら説明する。

【0218】端末 621、625 の処理能力  $c_1$  と  $c_2$  の関係に着目すると、端末 621 の処理能力  $c_1$  よりも端末 625 の処理能力  $c_2$  が低いので、端末 625 の処理能力  $c_2$  で処理可能なフレームレートに変換すればよいことになる。また、通信回線 622、624 の伝送速度  $b_1$  と  $b_2$  の関係に着目すると、通信回線 624 の伝送速度  $b_2$  は通信回線 622 の伝送速度  $b_1$  以下であるので、通信回線 624 の伝送速度  $b_2$  で伝送可能なフレームレートに変換すればよいことになる。

【0219】ここで、端末 625 の処理能力  $c_2$  で処理可能なフレーム数よりも、通信回線 624 の伝送速度  $b_2$  で伝送可能なフレーム数の方が多い場合、端末 625 の処理能力  $c_2$  で処理可能なフレーム数を超えるフレームを送信しても端末 625 ではデコードできないので、通信回線の使用効率が悪くなることになる。これを回避するために、端末 621 側から入力されたフレームレートの高いハイブリッド符号化された画像データは、端末 625 の処理能力  $c_2$  の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末 625 側に出力される。

【0220】また、端末 625 の処理能力  $c_2$  で処理可能なフレーム数よりも、通信回線 624 の伝送速度  $b_2$  で伝送可能なフレーム数の方が少ない場合、通信回線 624 の伝送速度  $b_2$  で伝送可能なフレーム数しか伝送することができないので、端末 621 側から入力されたフレームレートの高いハイブリッド符号化された画像データは、通信回線 624 の伝送速度  $b_2$  の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末 625 側に出力される。

【0221】また、さらに、端末 625 の処理能力  $c_2$  で処理可能なフレーム数と通信回線 624 の伝送速度  $b$

10

20

30

40

50



2で伝送可能なフレーム数が等しい場合、端末621側から入力されたフレームレートの高いハイブリッド符号化された画像データは、通信回線624の伝送速度 $b_2$ 、もしくは端末625の処理能力 $c_2$ の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末625側に出力される。

【0222】次に、フレームレートの高い画像データを、端末621側から入力し、通信回線の伝送速度が $b_1 < b_2$ となる関係で、かつ、端末の処理能力が、 $c_1 > c_2$ の関係となるときのフレームレート変換方式について図7を参照しながら説明する。

【0223】端末621、625の処理能力 $c_1$ と $c_2$ の関係に着目すると、端末621の処理能力 $c_1$ よりも端末625の処理能力 $c_2$ が低いので、端末625の処理能力 $c_2$ で処理可能なフレームレートに変換すればよいことになる。また、通信回線622、624の伝送速度 $b_1$ と $b_2$ の関係に着目すると、通信回線622の伝送速度 $b_1$ は通信回線624の伝送速度 $b_2$ より遅いので、通信回線622の伝送速度 $b_1$ で伝送可能なフレームレートに変換すればよいことになる。

【0224】ここで、端末625の処理能力 $c_2$ で処理可能なフレーム数よりも、通信回線622の伝送速度 $b_1$ で伝送可能なフレーム数の方が多い場合、端末625の処理能力 $c_2$ で処理可能なフレーム数を超えるフレームを伝送しても端末625ではデコードできないので、通信回線の使用効率が悪くなることになる。これを回避するために、端末621側から入力されたフレームレートの高いハイブリッド符号化された画像データは、端末625の処理能力 $c_2$ の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末625側に出力される。

【0225】また、端末625の処理能力 $c_2$ で処理可能なフレーム数よりも、通信回線622の伝送速度 $b_1$ で伝送可能なフレーム数の方が少ない場合、通信回線622の伝送速度 $b_1$ で伝送可能なフレーム数しか伝送することができないので、端末621側から入力されたフレームレートの高いハイブリッド符号化された画像データは、通信回線622の伝送速度 $b_1$ の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末625側に出力される。

【0226】また、さらに、端末625の処理能力 $c_2$ で処理可能なフレーム数と通信回線622の伝送速度 $b_1$ で伝送可能なフレーム数が等しい場合、端末621側から入力されたフレームレートの高いハイブリッド符号化された画像データは、通信回線622の伝送速度 $b_1$ 、もしくは端末625の処理能力 $c_2$ の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末625側に出力される。

力される。

【0227】次に、フレームレートの高い画像データを端末621側から入力し、通信回線の伝送速度が $b_1 > b_2$ となる関係で、かつ、端末の処理能力が、 $c_1 < c_2$ の関係となるときのフレームレート変換方式について図7を参照しながら説明する。

【0228】端末621、625の処理能力 $c_1$ と $c_2$ の関係に着目すると、端末625の処理能力 $c_2$ よりも端末621の処理能力 $c_1$ が低いので、端末621の処理能力 $c_1$ で処理可能なフレームレートをそのまま使用すればよいことになる。また、通信回線622、624の伝送速度 $b_1$ と $b_2$ の関係に着目すると、通信回線624の伝送速度 $b_2$ は通信回線622の伝送速度 $b_1$ よりも遅いので、通信回線624の伝送速度 $b_2$ で伝送可能なフレームレートに変換すればよいことになる。

【0229】ここで、端末621の処理能力 $c_1$ で処理可能なフレーム数よりも、通信回線624の伝送速度 $b_2$ で伝送可能なフレーム数の方が多い場合、端末621の処理能力 $c_1$ で処理可能なフレーム数を超えるフレームは通信回線624の伝送速度 $b_2$ を利用しても伝送できない。よって、端末621側から入力されたハイブリッド符号化された画像データは、フレームレート変換されずに、そのまま端末625側に出力される。

【0230】また、端末621の処理能力 $c_1$ で処理可能なフレーム数よりも、通信回線624の伝送速度 $b_2$ で伝送可能なフレーム数の方が少ない場合、通信回線624の伝送速度 $b_2$ で伝送可能なフレーム数しか伝送することができないので、端末621側から入力されたフレームレートの高いハイブリッド符号化された画像データは、通信回線624の伝送速度 $b_2$ の性能にあった、フレームレートの低いハイブリッド符号化された画像データにフレームレート変換されて端末625側に出力される。

【0231】また、さらに、端末621の処理能力 $c_1$ で処理可能なフレーム数と通信回線624の伝送速度 $b_2$ で伝送可能なフレーム数とが等しい場合、端末621側から入力されたハイブリッド符号化された画像データは、フレームレート変換されずに、そのまま端末625側に出力される。

【0232】図8は図7に示したフレームレート変換部の詳細な構成を示す図であり、上記実施の形態2から実施の形態5において、入力されたフレームレートの高い画像データで $(i+1)$ から $(i+n)$ 番目のフレームまでが、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータである場合の $(i+n)$ 番目のフレームの処理時のフレームレート変換方式を示すものである。

【0233】図8において、601は入力であり、フレームレートの高い画像データが入力される。

【0234】602は可変長復号化手段であり、ハイレ



リッド符号化データを可変長復号化することにより、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数に変換する。また、同時に、入力側端末の符号化装置で設定されたフレーム内／フレーム間識別を検出して、フレーム内／フレーム間 ( i n ) を設定し、また、入力側端末の符号化装置で設定された動き補償あり／なし識別を検出して、動き補償あり／なし ( i n ) を設定し、また、入力側端末の符号化装置で設定された量子化係数を検出して、量子化係数 ( i n ) を設定し、さらに、入力側端末の符号化装置で設定された動きベクトルを検出して、動きベクトル ( i n ) を設定する。

【 0 2 3 5 】 6 0 3 は逆量子化手段であり、上記設定された量子化係数 ( i n ) により、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を、動き補償を伴う予測誤差直交変換データ、動き補償を伴わない予測誤差直交変換データ、直交変換係数に変換する。

【 0 2 3 6 】 6 0 4 は予測誤差直交変換データ加算手段であり、動きベクトル・直交変換係数演算手段 6 0 8 にて動きベクトルを考慮して演算された直交変換係数、または画像メモリ 6 0 6 に記憶されている直交変換係数と、動き補償を伴う予測誤差直交変換データ、または動き補償を伴わない予測誤差直交変換データを加算する。

【 0 2 3 7 】 6 0 5 は直交変換係数選択手段であり、上記設定されたフレーム内／フレーム間 ( i n ) により制御される。すなわち、上記フレーム内／フレーム間 ( i n ) が「フレーム内」の時は、逆量子化手段 6 0 3 により逆量子化された直交変換係数が選択され、上記フレーム内／フレーム間 ( i n ) が「フレーム間」の時は、予測誤差直交変換データ加算手段 6 0 4 により算出された直交変換係数が選択され出力される。

【 0 2 3 8 】 6 0 6 は画像メモリであり、上記直交変換係数選択手段 6 0 5 により選択された直交変換係数を記憶するためのものである。

【 0 2 3 9 】 6 0 7 は動き補償あり／なし ( i n ) 判断手段であり、動き補償が「あり」の時は画像メモリ 6 0 6 に記憶されている直交変換係数を動きベクトル・直交変換係数演算手段 6 0 8 に出力する。動き補償が「なし」の時は画像メモリ 6 0 6 に記憶されている直交変換係数を予測誤差直交変換データ加算手段 6 0 4 に出力する。

【 0 2 4 0 】 6 0 8 は動きベクトル・直交変換係数演算手段であり、動きベクトル ( i n ) を用いて、動き補償あり／なし ( i n ) 判断手段 6 0 7 にて「あり」と判断されたときの画像メモリ 6 0 6 に記憶されていた直交変換係数を、マトリクス演算することにより、動きベクトルを考慮して演算された直交変換係数を算出する。マト

リクス演算については後述の方法を用いることとする。

【 0 2 4 1 】 6 0 9 は初期画像取込選択手段であり、後述する符号化制御・フレーム内／フレーム間判断手段 6 1 3 により設定された画像取込指示により、画像メモリ 6 0 6 に記憶されている直交変換係数を取り込み、後述する画像メモリ 6 1 0 に直交変換係数を出力する。

【 0 2 4 2 】 6 1 0 は画像メモリであり、初期画像取込選択手段 6 0 9 により選択された直交変換係数を記憶するためのものである。

【 0 2 4 3 】 6 1 1 はフレーム内／フレーム間 ( i n ) 判断手段であり、「フレーム内」の時は、画像メモリ 6 0 6 の直交変換係数を後段の符号化制御・フレーム内／フレーム間判断手段 6 1 3 へ出力し、「フレーム間」の時は、画像メモリ 6 0 6 の直交変換係数を後述する直交変換係数減算手段 6 1 2 へ出力する。

【 0 2 4 4 】 6 1 2 は直交変換係数減算手段であり、フレーム内／フレーム間 ( i n ) 判断手段 6 1 1 で、「フレーム間」と判断された画像メモリ 6 0 6 に記憶されていた直交変換係数から画像メモリ 6 1 0 の直交変換係数を減算する。

【 0 2 4 5 】 6 1 3 は符号化制御・フレーム内／フレーム間判断手段であり、フレームレート変換が行われていない時には、動き補償あり／なし ( i n ) をそのまま動き補償あり／なし ( o u t ) として出力し、フレームレート変換が行われている時には、動き補償あり／なし ( o u t ) は「なし」に設定して出力する。また、フレームレート変換を行う 1 つ前のフレームを画像メモリ 6 1 0 に記憶させるために、画像取込指示を出力する。また、フレーム内／フレーム間 ( i n ) を用いて、フレーム内／フレーム間 ( i n ) 判断手段 6 1 1 からの直交変換係数と、直交変換係数減算手段 6 1 2 からの動き補償を伴わない予測誤差直交変換データとの選択を行い、後述する量子化手段 6 1 4 へ出力する。また、量子化係数 ( i n ) を再設定し、量子化係数 ( o u t ) として後述する量子化手段 6 1 4 へ出力する。

【 0 2 4 6 】 6 1 4 は量子化手段であり、上記出力された量子化係数 ( o u t ) により、動き補償を伴わない予測誤差直交変換データ、直交変換係数を、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数にそれぞれ変換する。

【 0 2 4 7 】 6 1 5 はフレームレート変換制御手段であり、フレームレート変換情報より、フレームレート変換の継続／終了を判断する。そしてフレームレート変換終了後、可変長符号化手段 6 1 7 に、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を出力し、フレームレート変換が行われていない時は、可変長復号化手段 6 0 2 からの量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数をそのまま出力する。



【0248】616は動きベクトル選択手段であり、上記符号化制御・フレーム内／フレーム間判断手段613から出力された動き補償あり／なし(out)により動きベクトルの選択を行う。すなわち、動き補償あり／なし(out)が「あり」の時は、動きベクトル(in)を後述する可変長符号化手段617にそのまま出力し、「なし」の時には可変長符号化手段617には動きベクトル(in)を出力しない。

【0249】617は可変長符号化手段であり、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を可変長符号化することにより、ハイブリッド符号化データに変換する。また、同時に、フレーム内／フレーム間(in)により、出力側端末の復号化装置に対して設定されたフレーム内／フレーム間識別を設定し、動き補償あり／なし(out)、動きベクトル(in)により、出力側端末の復号化装置に対して動き補償あり／なし識別、動きベクトルを設定し、量子化係数(in)により、出力側端末の復号化装置に対して量子化係数を設定する。

【0250】618は出力であり、ここからフレームレートの低い画像データが出力される。

【0251】以下、フレームレート変換処理の詳細について説明する。入力される画像データを、0からi番目のフレームまではフレームレート変換なしとし、(i+1)から(i+n)番目のフレームまでを合成して、フレームレート変換を行い、(i+n+1)番目のフレーム以降のフレームは、フレームレート変換なしとして説明する。

【0252】フレームレート変換しない0～iフレームまでは、可変長復号化手段602、逆量子化手段603、予測誤差直交変換データ加算手段604、直交変換係数選択手段605、動き補償あり／なし(in)判断手段607、動きベクトル・直交変換係数演算手段608を経由して画像メモリ606にS(i)として記憶される。また、初期画像取込選択手段609を経由し、画像メモリ610にもS(i)として記憶される。また、可変長復号化手段602により可変長復号化されたQ(Emv(0))からQ(Emv(i))は、フレームレート変換制御手段615により選択されて、可変長符号化手段617により可変長符号化されて、V(Q(Emv(0)))からV(Q(Emv(i)))として出力される。

【0253】一方、フレームレート変換を行う(i+1)フレームは、可変長復号化手段602により、可変長復号化され、量子化された動き補償を伴う予測誤差直交変換データQ(Emv(i+1))に変換される。

【0254】そして、Q(Emv(i+1))は逆量子化手段603により、逆量子化され、動き補償を伴う予測誤差直交変換データEmv(i+1)に変換される。

【0255】このEmv(i+1)は、予測誤差直交変換データ加算手段604により、動き補償あり／なし(in)判断手段607で「あり」と判断される画像メモリ606に記憶されているS(i)を、動きベクトル・直交変換係数演算手段608で処理したS'(i)と加算され、直交変換係数選択手段605にて「フレーム間」と判断され、画像メモリ606にS(i+1)として記憶される。

【0256】(i+2)から(i+n)番目のフレームまでについても、上記(i+1)番目のフレームと同様の処理を繰り返すことで処理される。

【0257】そして、上記(i+n)フレームの処理が終了した時点で、画像メモリ606には、直交変換係数S(i+n)が記憶されており、画像メモリ610には直交変換係数S(i)が記憶されている。

【0258】上記直交変換係数S(i+n)は、フレーム内／フレーム間(in)判断手段611で、「フレーム間」と判断され、直交変換係数減算手段612により、S(i+n)-S(i)の処理が行われ、S(n)が算出される。ここで、S(n)は、動き補償を伴わない予測誤差直交変換データであり、(i+1)番目のフレームのデータとすると、E(i+1)=S(n)とすることができる。

【0259】上記動き補償を伴わない予測誤差直交変換データE(i+1)は符号化制御・フレーム内／フレーム間判断手段613にて、フレーム内／フレーム間(in)を用いることにより、「フレーム間」と判断され、選択される。フレームレート変換中は、動き補償あり／なし(out)は「なし」に設定される。画像取込指示は量子化手段614にE(i+1)が出力された後、出力される。

【0260】E(i+1)は量子化手段614により量子化されて、量子化された動き補償を伴わない予測誤差直交変換データQ(E(i+1))に変換される。

【0261】この量子化された動き補償を伴わない予測誤差直交変換データQ(E(i+1))は可変長符号化手段617により、可変長符号化され、量子化され可変長符号化された動き補償を伴わない予測誤差直交変換データV(Q(E(i+1)))として出力される。

【0262】一方、フレームレート変換しない(i+n+1)番目のフレームは、可変長復号化手段602、逆量子化手段603、予測誤差直交変換データ加算手段604、直交変換係数選択手段605、動き補償あり／なし(in)判断手段607、動きベクトル・直交変換係数演算手段608を経由して、画像メモリ606に、直交変換係数S(i+n+1)として記憶される。また、初期画像取込選択手段609を経由して、画像メモリ610にもS(i+n+1)として記憶される。また、可変長復号化手段602により可変長復号化されて得られたQ(Emv(i+n+1))は、フレームレート変換



制御手段 6 1 5 により選択されて、 $Q(E_{mv}(i+2))$  となり、さらに可変長符号化手段 6 1 7 により可変長符号化されて、 $V(Q(E_{mv}(i+2)))$  として出力される。そして、 $(i+n+1)$  より後のフレームについても上記  $(i+n+1)$  番目のフレームと同様の処理が行われる。

【0 2 6 3】以上のように、本実施の形態によれば、フレームレート変換部 6 2 3 で、フレームレートの高いハイブリッド符号化された画像データを、フレームレートの低いハイブリッド符号化された画像データに変換する際に、通信回線の伝送速度と端末の処理能力を考慮し、最も律速となる処理速度に合わせてフレームレートを

変換するようにしたから、通信回線の使用効率が悪化することがなくなる。  
【0 2 6 4】また、ハイブリッド符号化された画像データの中の任意の開始フレーム、任意の合成フレーム枚数を制御することにより、動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータに対して、該データのみを選択的にフレームレート変換を行うことができ、演算量が大幅に減少し、高速で負荷の少ないフレームレート変換を行うことができる。

【0 2 6 5】実施の形態 7. 以下、本発明の請求項 7 に対応する実施の形態 7. に係るフレームレート変換方式を利用するフレームレート変換装置について、図面を参照しながら説明する。

【0 2 6 6】図 9 は上記実施の形態 1 から 6 におけるフレームレート変換方式を利用したフレームレート変換装置を示すものである。

【0 2 6 7】図 9 において、7 0 1 は入力端子であり、処理能力の高い端末からのフレームレートの高い画像データまたは、伝送速度の速い通信回線からのフレームレートの高い画像データが入力される。

【0 2 6 8】7 0 2 は可変長復号化手段であり、ハイブリッド符号化データを可変長復号化することにより、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数に変換する。また、同時に、入力側端末の符号化装置で設定されたフレーム内／フレーム間識別を検出して、フレーム内／フレーム間  $(i_n)$  を設定し、また、入力側端末の符号化装置で設定された動き補償あり／なし識別を検出して、動き補償あり／なし  $(i_n)$  を設定し、また、入力側端末の符号化装置で設定された量子化係数を検出して、量子化係数  $(i_n)$  を設定し、さらに、入力側端末の符号化装置で設定された動きベクトルを検出して、動きベクトル  $(i_n)$  を設定する。

【0 2 6 9】7 0 3 は逆量子化手段であり、上記設定された量子化係数  $(i_n)$  により、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償

を伴わない予測誤差直交変換データ、量子化された直交変換係数を、動き補償を伴う予測誤差直交変換データ、動き補償を伴わない予測誤差直交変換データ、直交変換係数にそれぞれ変換する。

【0 2 7 0】7 0 4 は予測誤差直交変換データ加算手段であり、動きベクトル・直交変換係数演算手段 7 0 8 にて動きベクトルを考慮して演算された直交変換係数、または画像メモリ 7 0 6 に記憶されている直交変換係数と、動き補償を伴う予測誤差直交変換データ、または動き補償を伴わない予測誤差直交変換データとを加算する。

【0 2 7 1】7 0 5 は直交変換係数選択手段であり、上記設定されたフレーム内／フレーム間  $(i_n)$  により制御される。すなわち、上記フレーム内／フレーム間  $(i_n)$  が「フレーム内」の時は、逆量子化手段 7 0 3 により逆量子化された直交変換係数が選択され、上記フレーム内／フレーム間  $(i_n)$  が「フレーム間」の時は、予測誤差直交変換データ加算手段 7 0 4 により算出された直交変換係数が選択され出力される。

【0 2 7 2】7 0 6 は画像メモリであり、上記直交変換係数選択手段 7 0 5 により選択された直交変換係数を記憶するためのものである。

【0 2 7 3】7 0 7 は動き補償あり／なし  $(i_n)$  判断手段であり、動き補償が「あり」の時は画像メモリ 7 0 6 に記憶されている直交変換係数を動きベクトル・直交変換係数演算手段 7 0 8 に出力する。動き補償が「なし」の時は画像メモリ 7 0 6 に記憶されている直交変換係数を予測誤差直交変換データ加算手段 7 0 4 に出力する。

【0 2 7 4】7 0 8 は動きベクトル・直交変換係数演算手段であり、動きベクトル  $(i_n)$  を用いて、動き補償あり／なし  $(i_n)$  判断手段 7 0 7 にて「あり」と判断された画像メモリ 7 0 6 に記憶されていた直交変換係数を、マトリクス演算することにより、動きベクトルを考慮して演算された直交変換係数を算出する。マトリクス演算については後述の方法を用いる。

【0 2 7 5】7 0 9 は初期画像取込選択手段であり、後述する符号化制御・フレーム内／フレーム間判断手段 7 1 3 により設定された画像取込指示により、画像メモリ 7 0 6 に記憶されているの直交変換係数を取り込み、後述する画像メモリ 7 1 0 に直交変換係数を出力する。

【0 2 7 6】7 1 0 は画像メモリであり、初期画像取込選択手段 7 0 9 により選択された直交変換係数を記憶するためのものである。

【0 2 7 7】7 1 1 はフレーム内／フレーム間  $(i_n)$  判断手段であり、「フレーム内」の時は、画像メモリ 7 0 6 の直交変換係数を後段の符号化制御・フレーム内／フレーム間判断手段 7 1 3 へ出力し、「フレーム間」の時は、画像メモリ 7 0 6 の直交変換係数を後述する直交変換係数減算手段 7 1 2 へ出力する。



【0278】712は直交変換係数減算手段であり、フレーム内／フレーム間 (i n) 判断手段711で、「フレーム間」と判断されたときに画像メモリ706に記憶されていた直交変換係数から、画像メモリ710の直交変換係数を減算する。

【0279】713は符号化制御・フレーム内／フレーム間判断手段であり、フレームレート変換が行われていない時には、動き補償あり／なし (i n) をそのまま動き補償あり／なし (o u t) として出力し、フレームレート変換が行われている時には、動き補償あり／なし (o u t) は「なし」に設定して出力する。また、フレームレート変換を行う1つ前のフレームを画像メモリ710に記憶させるために、画像取込指示を出力する。また、フレーム内／フレーム間 (i n) を用いて、フレーム内／フレーム間 (i n) 判断手段711からの直交変換係数と、直交変換係数減算手段712からの動き補償を伴わない予測誤差直交変換データとの選択を行い、量子化手段714へ出力する。また、量子化係数 (i n) を再設定し、量子化係数 (o u t) として後述する量子化手段714へ出力する。

【0280】714は量子化手段であり、上記出力された量子化係数 (o u t) により、動き補償を伴わない予測誤差直交変換データ、直交変換係数を、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数にそれぞれ変換する。

【0281】715はフレームレート変換制御手段であり、フレームレート変換情報よりフレームレート変換の継続／終了を判断する。そして、フレームレート変換終了後、可変長符号化手段717に、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を出力し、フレームレート変換が行われない時は、可変長復号化手段702からの量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数をそのまま出力する。

【0282】716は動きベクトル選択手段であり、上記符号化制御・フレーム内／フレーム間判断手段713から出力された動き補償あり／なし (o u t) により、動きベクトルの選択を行う。すなわち、動き補償あり／なし (o u t) が「あり」の時は、動きベクトル (i n) を可変長符号化手段717にそのまま出力し、「なし」の時には、可変長符号化手段717には動きベクトル (i n) を出力しない。

【0283】717は可変長符号化手段であり、量子化された動き補償を伴う予測誤差直交変換データ、量子化された動き補償を伴わない予測誤差直交変換データ、量子化された直交変換係数を可変長符号化することにより、ハイブリッド符号化データに変換する。また、同時に、フレーム内／フレーム間 (i n) により、出力側端

ム間識別を設定し、動き補償あり／なし (o u t) 、動きベクトル (i n) により、出力側端末の復号化装置に対して動き補償あり／なし識別、動きベクトルを設定し、量子化係数 (i n) により、出力側端末の復号化装置に対して量子化係数を設定する。

【0284】718は出力であり、処理能力の低い端末へのフレームレートの低い画像データまたは、伝送速度の遅い通信回線へのフレームレートの低い画像データが出力される。

【0285】以下に、上記実施の形態1～7で用いられる動きベクトル・直交変換係数演算手段(108, 208, 308, 408, 508, 608, 708)により、動きベクトルを考慮して演算された直交変換係数を算出するためのマトリクス演算について説明する。

【0286】図11は、上記動きベクトル・直交変換係数演算手段(108, 208, 308, 408, 508, 608, 708)の詳細を説明するフローチャートを示す図で、1001はブロック探索手段、1002は直交変換係数演算手段、1003は直交変換係数加算手段を示す。

【0287】ここで、動きベクトル・直交変換係数演算手段(108, 208, 308, 408, 508, 608, 708)の詳細を説明するために、以下のことを定義しておく。

【0288】動きベクトル・直交変換係数演算手段(108, 208, 308, 408, 508, 608, 708)は、動き補償ありの場合、画像メモリ(106, 206, 306, 406, 506, 606, 706)に記憶されていた直交変換係数と動きベクトル (i n) を取り込む。

【0289】二次元直交変換の処理をF、二次元直交変換を一次元直交変換に分解した時の変換行列をT、転置行列をtで表す。

【0290】入力された画像データで動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化されたデータを逆直交変換した画像ブロックをM、画像メモリ(106, 206, 306, 406, 506, 606, 706)に記憶されていた直交変換係数を逆直交変換した画像ブロックをAとする。AはMの参照ブロックである。画像メモリ(106, 206, 306, 406, 506, 606, 706)に記憶されていたデータは直交変換係数であるのでブロック境界で分割されており、Aは最大で4分割される場合がある。この4分割された場合のブロックを左上ブロックをA1、右上ブロックをA2、左下ブロックをA3、右下ブロックをA4とする。A1, A2, A3, A4におけるAとの共通部分をMに対応する位置に移動し、移動した部分以外を0とした画像ブロックをそれぞれB1, B2, B3, B4とする。図12は、画像ブロックM、A、A1, A2, A3, A4、B1, B2, B3, B4を示す



ものである。図 12 の A1, A2, A3, A4 と B1, B2, B3, B4 の同じ網掛け部分はそれぞれの値は等しい。また、A と M に使用されている記号は位置関係を示しているだけであり、それぞれの値は異なる。

【0291】図 11 の 1001 はブロック探索手段であり、動きベクトル (in) を用いることにより、A が探索される。

【0292】図 11 の 1002 は直交変換係数演算手段であり、上記探索された A から、F (A1), F (A2), F (A3), F (A4) を求める。ここで、F (A1), F (A2), F (A3), F (A4) は、画像メモリ (106, 206, 306, 406, 506, 606, 706) に記憶されていた直交変換係数の画像ブロックと等しい。後述するマトリクス演算を利用して、F (B1), F (B2), F (B3), F (B4) を求める。以下に、マトリクス演算の方法を示す。この演算を行うために用いるマトリクスを図 13, 図 14 に示す。このマトリクスを S 行列とする。また、ここではブロックの大きさが 8×8 である場合について説明する。図 15, 図 16, 図 17, 18 は、S 行列を掛けた時の 8×8 のブロックの移動について示したものである。m=1~7 とすると、Sdlm を 8×8 のブロックに対して左から掛けると図 15 に示すように、8×8 の

ブロックは m 行分下に移動され、上から m 行目まで全て 0 に変換される。Sdlm を 8×8 のブロックに対して右から掛けると図 16 に示すように、8×8 のブロックは m 列分左に移動され、右から m 列目まで全て 0 に変換される。Surm を 8×8 のブロックに対して左から掛けると図 17 に示すように、8×8 のブロックは m 行分上に移動され、下から m 行目まで全て 0 に変換される。Surm を 8×8 のブロックに対して右から掛けると図 18 に示すように、8×8 のブロックは m 列分右に移動され、左から m 列目まで全て 0 に変換される。

【0293】次に F (A1) を F (B1) に変換する方法については図面を参照しながら説明する。

【0294】図 19 は A1 を B1 に変換する方法の説明を示したものである。A, A1, B1 が 8×8 のブロックであり、A1 と A の共通部分の大きさを a 行 b 列とする。A1 を上に (8-a) 行分、左に (8-b) 列分移動し、右から 8-a 行目、下から 8-b 列目まで 0 に変換すればよいことになる。この変換は、A1 に対して Sur (8-a) を左から掛け、この後、Sdl (8-b) を右から掛けることにより行われる。これを式で表すと、

$Sur (8-a) \times A1 \times Sdl (8-b) = B1$  となる。F (B1) を求めるには

$$\begin{aligned} & F (B1) \\ &= F (Sur (8-a) \times A1 \times Sdl (8-b)) \\ &= T Sur (8-a) \times A1 \times Sdl (8-b) t T \\ &= T Sur (8-a) T t \times T A1 T t \times T Sdl (8-b) t T \\ &= F (Sur (8-a)) \times F (A1) \times F (Sdl (8-b)) \end{aligned}$$

とすればよい。これにより、S 行列自体は必要でないということが分かり、F (S 行列) については予め計算し、準備しておけば、F (A1) から F (B1) を求めることが容易になる。F (B2), F (B3), F (B4) も同様に求めることができる。

【0295】図 11 の 1003 は直交変換係数加算手段であり、F (B1), F (B2), F (B3), F (B4) を加算し、F (A) を求める。この処理の詳細を示すと、直交変換は線形の操作であるので、F (B1), F (B2), F (B3), F (B4) を加算すると、 $F (B1) + F (B2) + F (B3) + F (B4) = F (B1 + B2 + B3 + B4)$

となり、ここで、B1~B4 の網掛け部分と A1~A4 の網掛け部分が等しいので

$B1 + B2 + B3 + B4 = A$

であるので、

$F (B1 + B2 + B3 + B4) = F (A)$

よって、

$F (B1) + F (B2) + F (B3) + F (B4) = F (A)$

となる。

【0296】この操作により求めた F (A) はフレー

ム単位でまとめられることにより、S' (i) となる。

【0297】以上のように、本実施の形態によれば、逆直交変換、動き検出、直交変換の処理を省略することにより演算量を減らし、従来の技術に比べ大幅な負荷の軽減と高速化を図り、通信回線の使用効率の向上をはかることができ、また、端末の処理能力に対応したフレームレートの画像データを生成することができるという効果がある。

【0298】

【発明の効果】以上のように、本発明の請求項 1 に係るフレームレート変換方式によれば、端末間でハイブリッド符号化された画像データを通信する場合において、通信回線の伝送速度によってフレームレートを変換するだけではなく、端末の処理能力を考慮することにより、入力されたフレームレートの高い画像データを、端末の処理能力に応じたフレームレートに変換し、フレームレートの低い画像データとして出力するようにしたので、出力において、通信回線で伝送可能なフレーム数よりも、端末で処理できるフレーム数の方が少ない場合は、端末の処理能力を考慮してフレームレートの変換が行われ、端末で処理できないフレーム数のハイブリッド符号化された画像データが通信回線に出力されることがなくな



り、通信回線の使用効率を向上させることができる効果がある。

【0299】また、本発明の請求項2に係るフレームレート変換方式によれば、フレームレートの高い動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データを入力し、逆直交変換、動き検出、直交変換の処理を省略することにより演算量を減らし、従来の技術に比べ大幅な負荷の軽減と高速化を図ることができ、また、フレームレートの低い動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データを生成することができるフレームレート変換方式を提供できるという効果がある。

【0300】また、本発明の請求項3に係るフレームレート変換方式によれば、フレームレートの高い動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データとの両方が含まれる画像データを入力し、逆直交変換、動き検出、直交変換の処理を省略することにより演算量を減らし、従来の技術に比べ大幅な負荷の軽減と高速化を図ることができ、また、フレームレートの低い動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データを生成することができるフレームレート変換方式を提供できるという効果がある。

【0301】また、本発明の請求項4に係るフレームレート変換方式によれば、フレームレートの高い動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データを入力し、逆直交変換、動き検出、直交変換の処理を省略することにより演算量を減らし、従来の技術に比べ大幅な負荷の軽減と高速化を図ることができ、また、フレームレートの低い動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データを生成することができるフレームレート変換方式を提供できるという効果がある。

【0302】また、本発明の請求項5に係るフレームレート変換方式によれば、フレームレートの高い動き補償を伴うフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、動き補償を伴わないフレーム間予測誤差に対して直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの3つが含まれる画像データを入力し、逆直交変換、動き検出、直交変換の処理を省略することにより演算量を減らし、従来の技術に比べ大幅な負荷の軽減と高速化を図ることができ、また、フレームレートの低い動き補償を伴わないフレーム間予測誤差に対して

直交変換を行うハイブリッド符号化された画像データと、フレーム内符号化された画像データとの両方が含まれる画像データを生成することができるフレームレート変換方式を提供できるという効果がある。

【0303】また、本発明の請求項6に係るフレームレート変換方式によれば、端末間でハイブリッド符号化された画像データを通信する場合において、通信回線の伝送速度によってフレームレートを変換するだけではなく、端末の処理能力を考慮することにより、入力されたフレームレートの高い画像データを、端末の処理能力に応じたフレームレートに、上記請求項2ないし請求項5のいずれかに記載のフレームレート変換方式を用いて変換し、逆直交変換、動き検出、直交変換の処理を省略することにより演算量を減らし、従来の技術に比べ大幅な負荷の軽減と高速化を図ることができ、フレームレートの低い画像データとして出力するようにしたので、出力において、通信回線で伝送可能なフレーム数よりも、端末で処理できるフレーム数の方が少ない場合は、端末の処理能力を考慮してフレームレートの変換が行われ、端末で処理できないフレーム数のハイブリッド符号化された画像データが通信回線に出力されることがなくなり、通信回線の使用効率を向上させることができる効果がある。

【0304】また、本発明の請求項7に係るフレームレート変換装置によれば、上記請求項1ないし請求項6のいずれかに記載のフレームレート変換方式を利用して、逆直交変換、動き検出、直交変換の処理を省略することにより演算量を減らし、従来の技術に比べ大幅な負荷の軽減と高速化を図り、通信回線の使用効率の向上をはかることができ、また、端末の処理能力に対応したフレームレートの画像データを生成することができるという効果がある。

#### 【図面の簡単な説明】

【図1】本発明の請求項1に係るフレームレート変換方式の構成を説明するための図である。

【図2】上記請求項1に係るフレームレート変換方式によるフレームレート変換処理を説明するためのフローチャートを示す図である。

【図3】本発明の請求項2に係るフレームレート変換方式を説明するためのフローチャートを示す図である。

【図4】本発明の請求項3に係るフレームレート変換方式を説明するためのフローチャートを示す図である。

【図5】本発明の請求項4に係るフレームレート変換方式を説明するためのフローチャートを示す図である。

【図6】本発明の請求項5に係るフレームレート変換方式を説明するためのフローチャートを示す図である。

【図7】本発明の請求項6に係るフレームレート変換方式の構成を説明するための図である。

【図8】上記請求項6に係るフレームレート変換方式によるフレームレート変換処理を説明するためのフローチャート



ャートを示す図である。

【図 9】本発明の請求項 7 に係るフレームレート変換装置の構成を示す図である。

【図 10】従来のレート変換画像符号化装置の構成を示す図である。

【図 11】動きベクトル・直交変換係数演算手段の詳細を説明するフローチャートを示す図である。

【図 12】画像ブロック M、A、A1、A2、A3、A4、B1、B2、B3、B4 を示す図である。

【図 13】動きベクトルを考慮して演算された直交変換係数を算出するためのマトリクス演算方法にて用いられるマトリクスを示す図である。

【図 14】動きベクトルを考慮して演算された直交変換係数を算出するためのマトリクス演算方法にて用いられるマトリクスを示す図である。

【図 15】Sd1m 行列を左から掛けた時の 8×8 のブロックの移動について示す図である。

【図 16】Sd1m 行列を右から掛けた時の 8×8 のブロックの移動について示す図である。

【図 17】Surm 行列を左から掛けた時の 8×8 のブロックの移動について示す図である。

【図 18】Surm 行列を右から掛けた時の 8×8 のブロックの移動について示す図である。

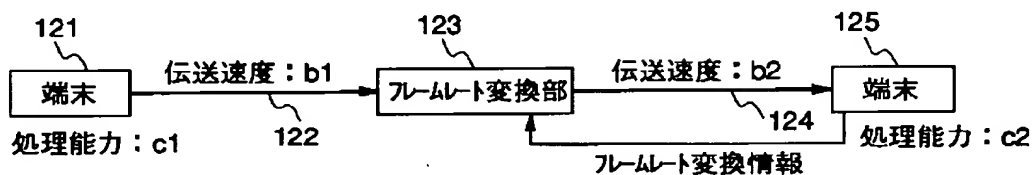
【図 19】A1 を B1 に変換する方法の説明を示す図である。

【図 20】従来のフレームレート変換方式の構成を示す図である。

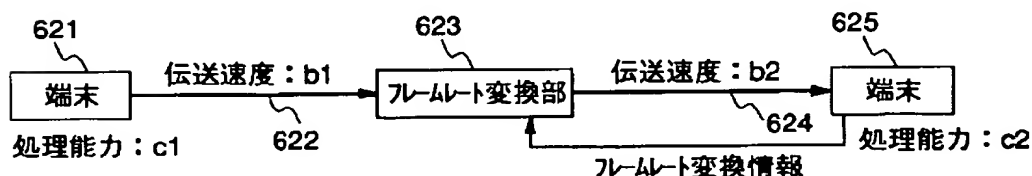
【符号の説明】

- 104 予測誤差直交変換データ加算手段
- 112 直交変換係数減算手段
- 204 予測誤差直交変換データ加算手段
- 212 直交変換係数減算手段
- 304 予測誤差直交変換データ加算手段
- 312 直交変換係数減算手段
- 404 予測誤差直交変換データ加算手段
- 412 直交変換係数減算手段
- 504 予測誤差直交変換データ加算手段
- 512 直交変換係数減算手段
- 604 予測誤差直交変換データ加算手段
- 612 直交変換係数減算手段
- 704 予測誤差直交変換データ加算手段
- 705 直交変換係数選択手段
- 707 動き補償あり／なし (in) 判断手段
- 709 初期画像取込選択手段
- 711 フレーム内／フレーム間 (in) 判断手段
- 712 直交変換係数減算手段
- 715 フレームレート変換制御手段
- 919 加算手段
- 921 スイッチ手段
- 922 セレクタ
- 924 加算手段
- 927 セレクタ
- 928 減算手段
- 931 セレクタ
- 934 加算手段
- 937 セレクタ

【図 1】



【図 7】

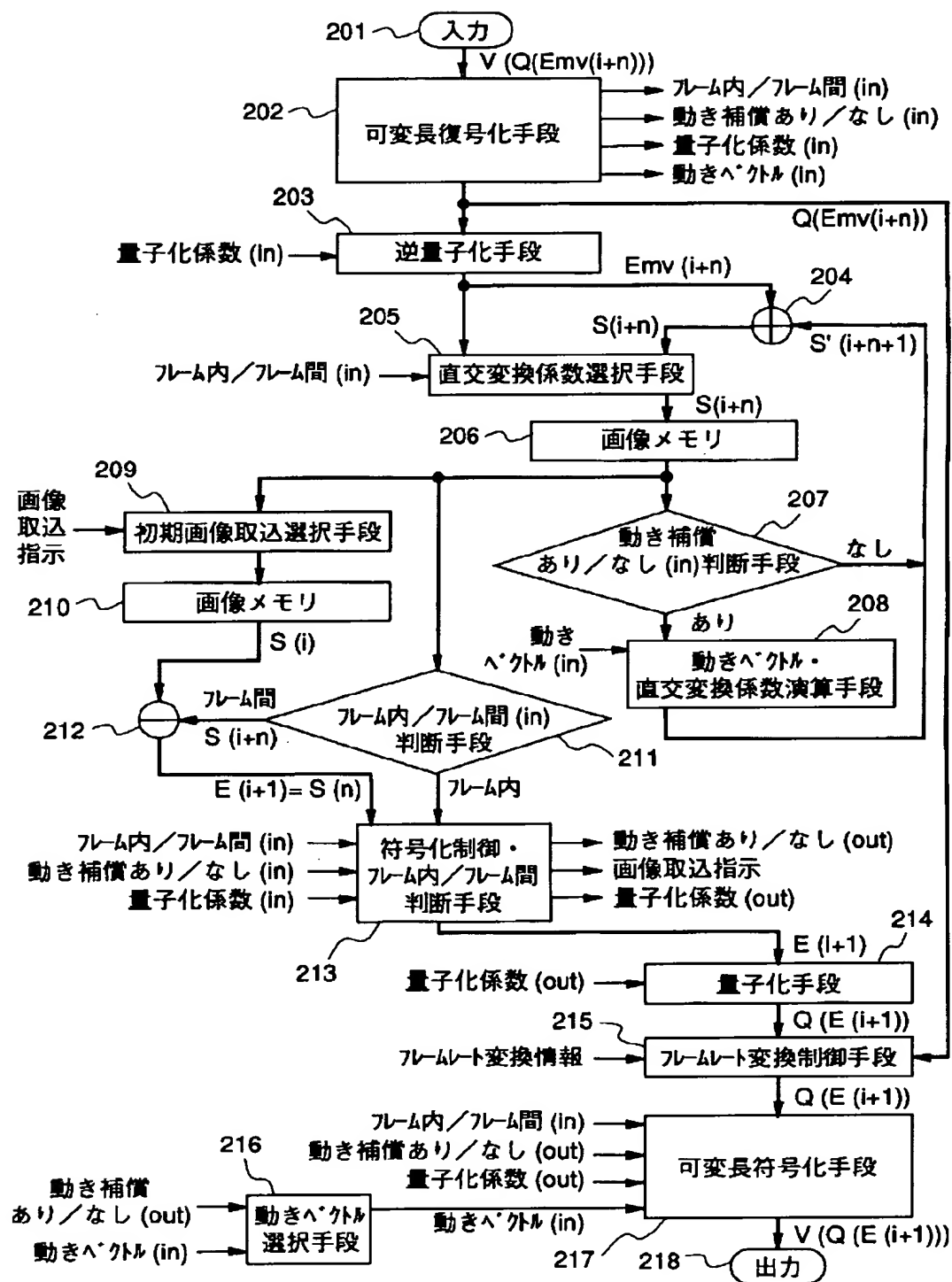








【図 3】

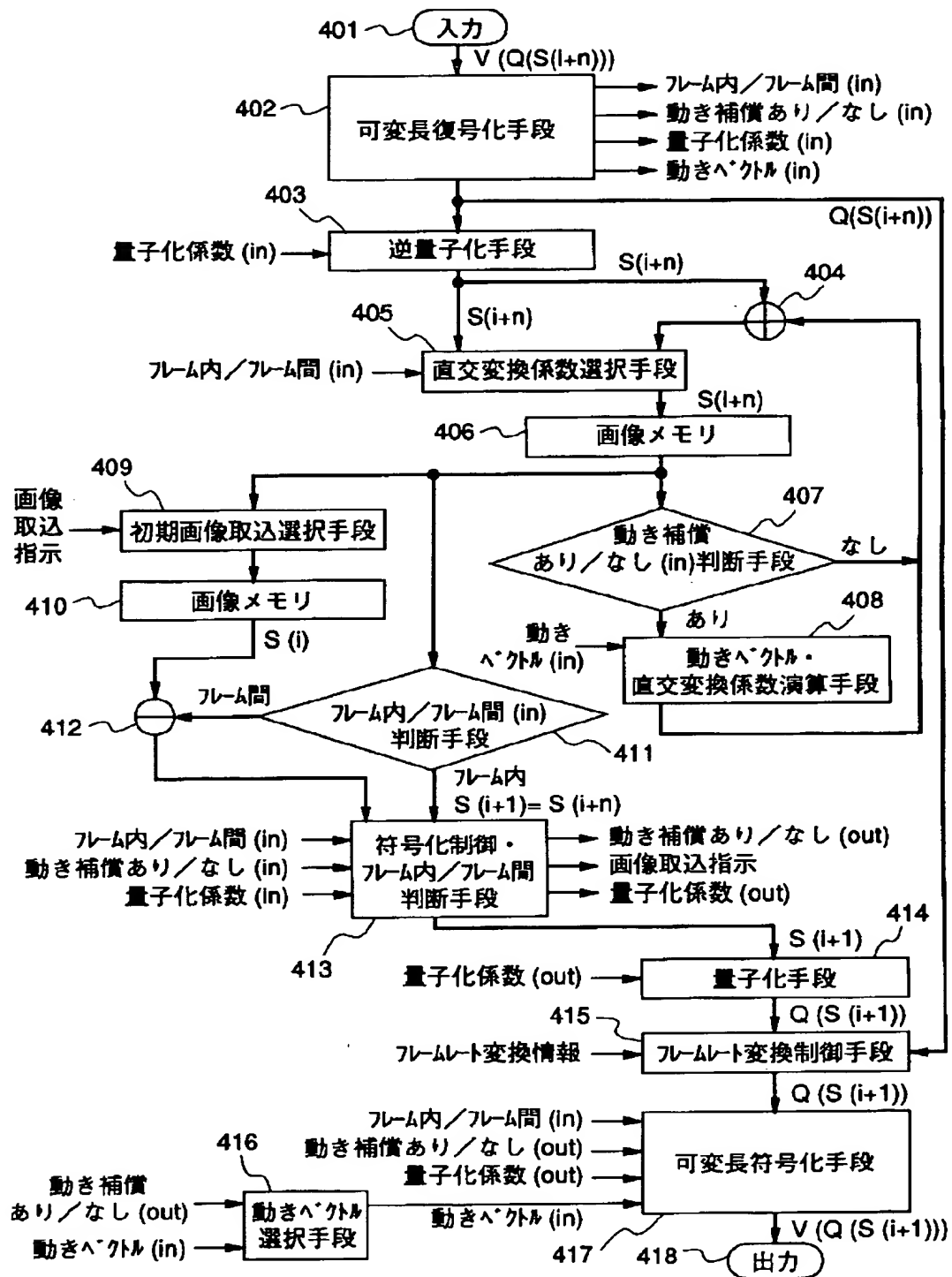






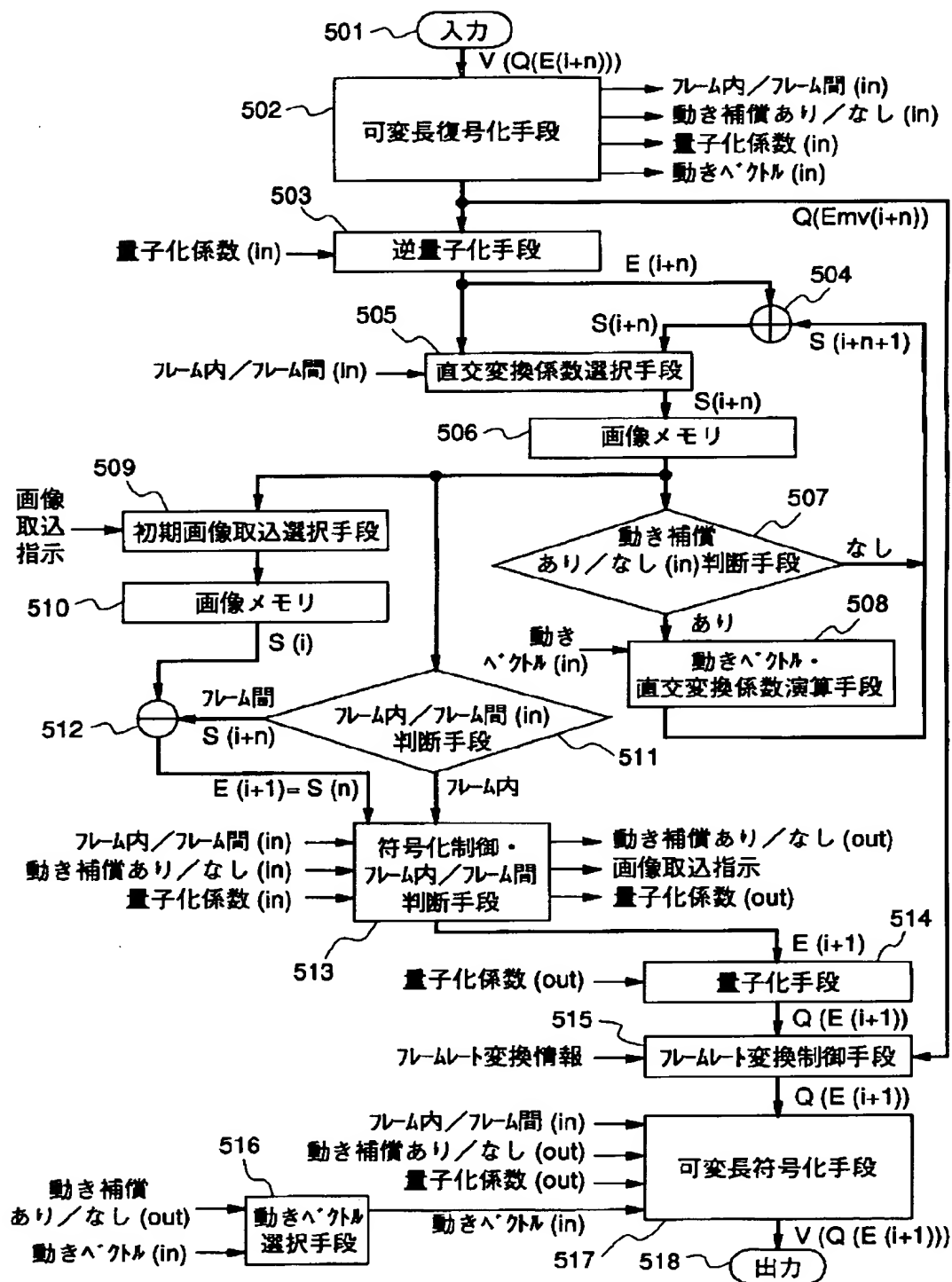


【図 5】



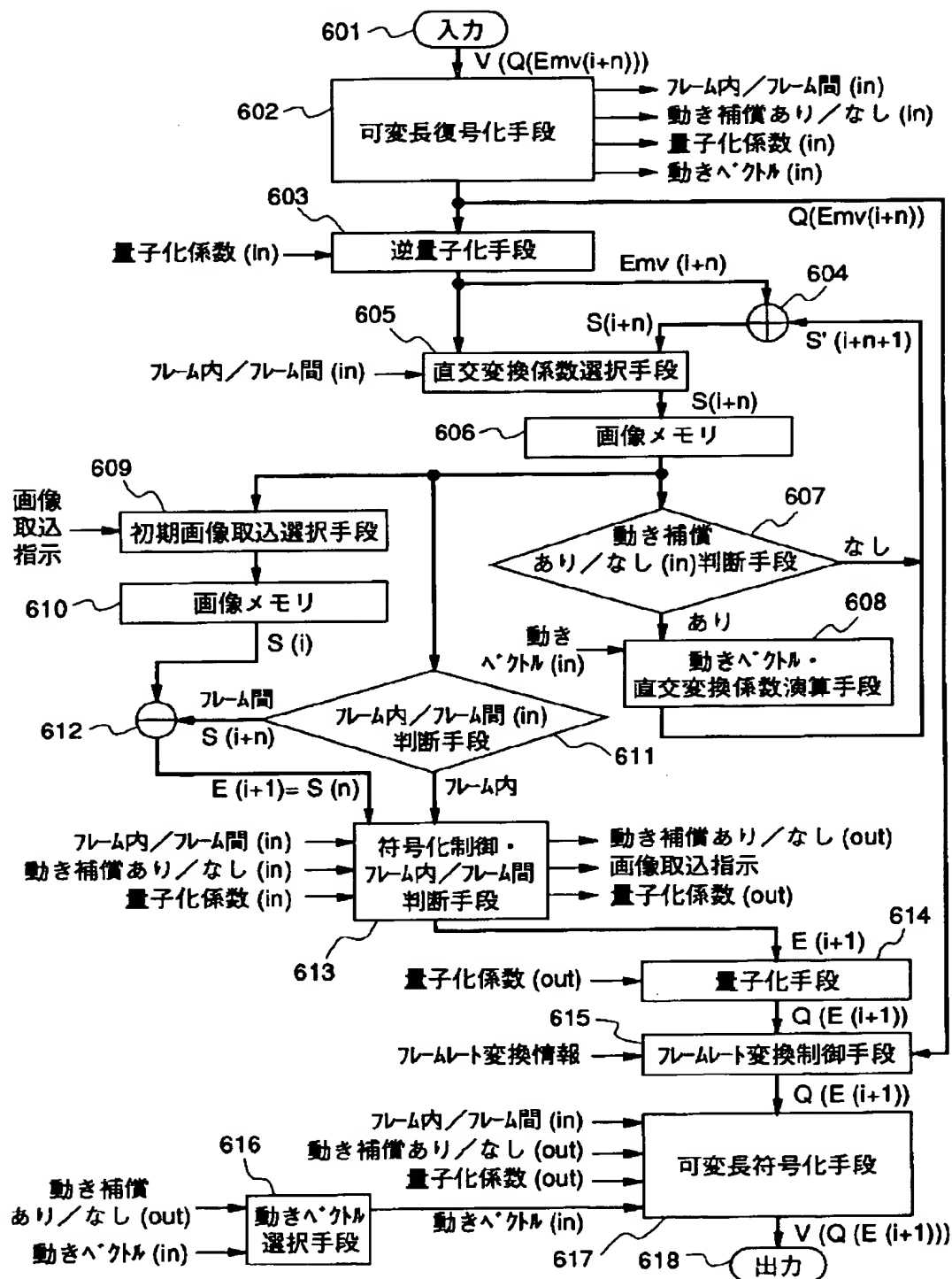


【図 6】



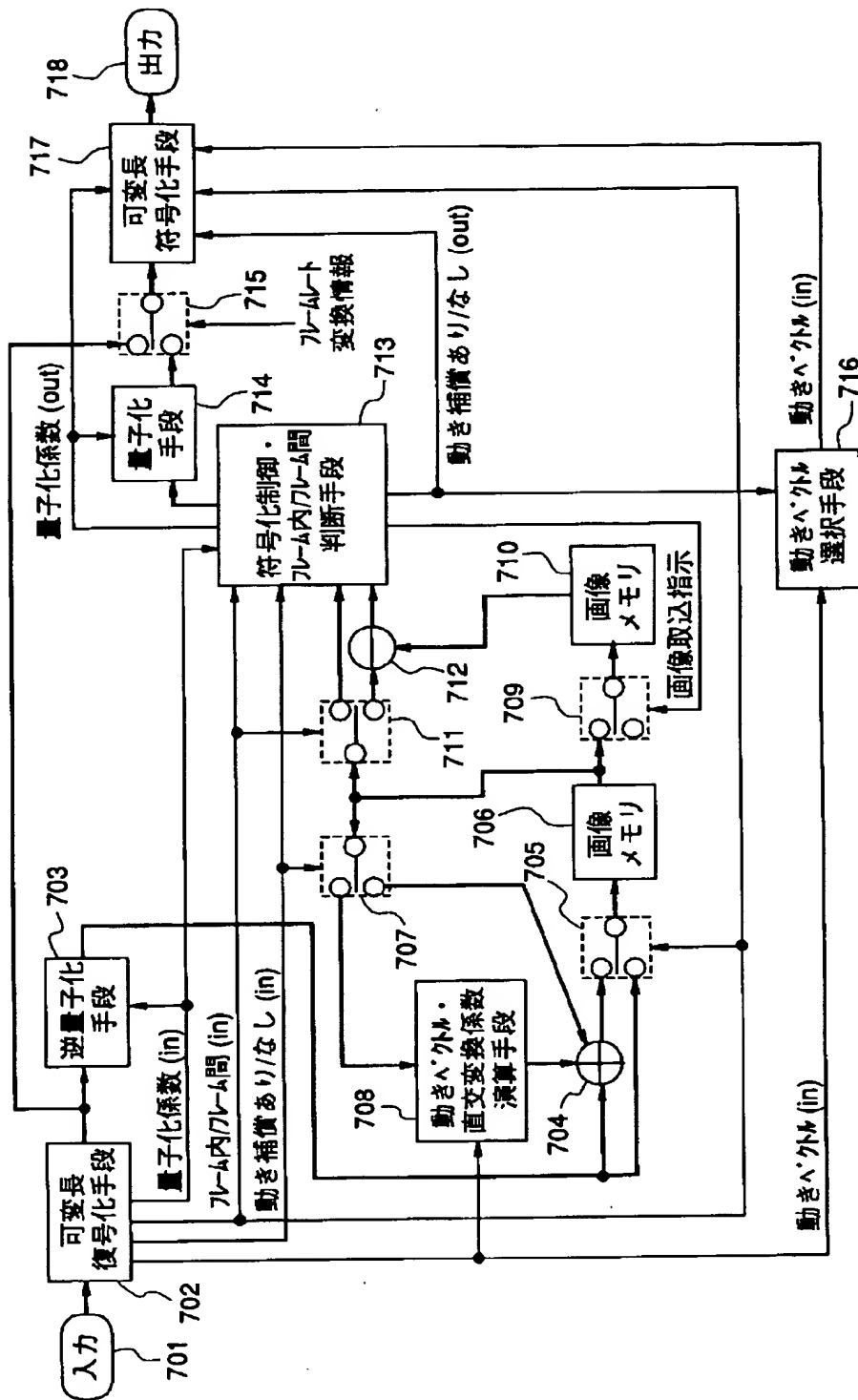


【图 8】





【図 9】



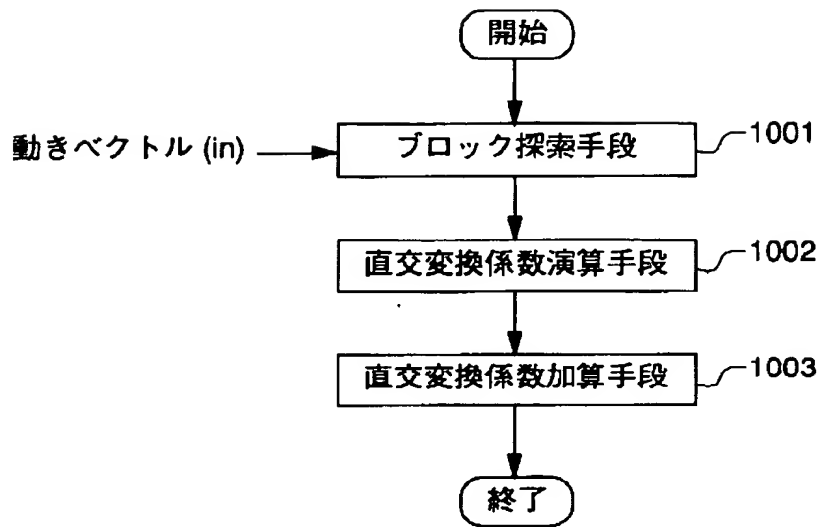


The diagram illustrates a video processing system with the following components and signal flow:

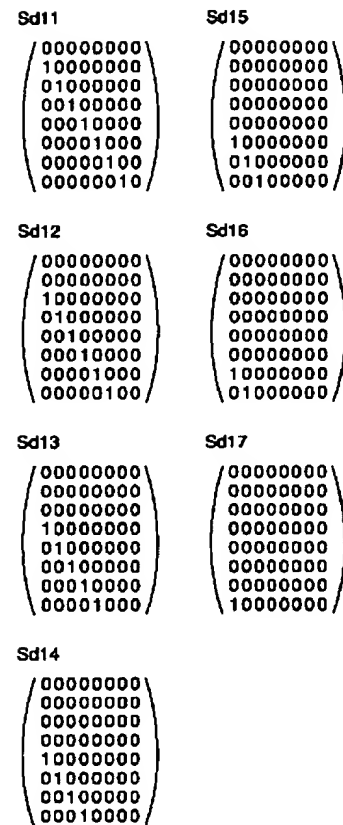
- Input (901):** The starting point of the signal.
- 可変長復号化手段 (902):** Variable length decoding unit.
- 逆量子化手段 (903):** Inverse quantization unit.
- 量子化係数 (904):** Quantization coefficients.
- 7フレーム内7フレーム間 (905):** 7-frame intra/inter-frame processing unit.
- 直交変換手段 (906):** Orthogonal transformation unit.
- 符号化制御手段 (907):** Encoding control unit.
- 量子化係数 (908):** Quantization coefficients.
- 可変長符号化手段 (909):** Variable length encoding unit.
- 出力 (910):** The final output of the system.
- 逆直交変換手段 (911):** Inverse orthogonal transformation unit.
- 量子化手段 (912):** Quantization unit.
- 逆量子化手段 (913):** Inverse quantization unit.
- 7フレーム内7フレーム間 (914):** 7-frame intra/inter-frame processing unit.
- 直交変換手段 (915):** Orthogonal transformation unit.
- 符号化制御手段 (916):** Encoding control unit.
- 量子化係数 (917):** Quantization coefficients.
- 可変長符号化手段 (918):** Variable length encoding unit.
- 出力 (919):** The final output of the system.
- 逆直交変換手段 (920):** Inverse orthogonal transformation unit.
- 量子化手段 (921):** Quantization unit.
- 逆量子化手段 (922):** Inverse quantization unit.
- 7フレーム内7フレーム間 (923):** 7-frame intra/inter-frame processing unit.
- 直交変換手段 (924):** Orthogonal transformation unit.
- 符号化制御手段 (925):** Encoding control unit.
- 量子化係数 (926):** Quantization coefficients.
- 可変長符号化手段 (927):** Variable length encoding unit.
- 出力 (928):** The final output of the system.
- 逆直交変換手段 (929):** Inverse orthogonal transformation unit.
- 量子化手段 (930):** Quantization unit.
- 逆量子化手段 (931):** Inverse quantization unit.
- 7フレーム内7フレーム間 (932):** 7-frame intra/inter-frame processing unit.
- 直交変換手段 (933):** Orthogonal transformation unit.
- 符号化制御手段 (934):** Encoding control unit.
- 量子化係数 (935):** Quantization coefficients.
- 可変長符号化手段 (936):** Variable length encoding unit.
- 出力 (937):** The final output of the system.
- 逆直交変換手段 (938):** Inverse orthogonal transformation unit.
- 量子化手段 (939):** Quantization unit.
- 逆量子化手段 (940):** Inverse quantization unit.
- 7フレーム内7フレーム間 (941):** 7-frame intra/inter-frame processing unit.
- 直交変換手段 (942):** Orthogonal transformation unit.
- 符号化制御手段 (943):** Encoding control unit.
- 量子化係数 (944):** Quantization coefficients.
- 可変長符号化手段 (945):** Variable length encoding unit.
- 出力 (946):** The final output of the system.
- 逆直交変換手段 (947):** Inverse orthogonal transformation unit.
- 量子化手段 (948):** Quantization unit.
- 逆量子化手段 (949):** Inverse quantization unit.
- 7フレーム内7フレーム間 (950):** 7-frame intra/inter-frame processing unit.
- 直交変換手段 (951):** Orthogonal transformation unit.
- 符号化制御手段 (952):** Encoding control unit.
- 量子化係数 (953):** Quantization coefficients.
- 可変長符号化手段 (954):** Variable length encoding unit.
- 出力 (955):** The final output of the system.
- 逆直交変換手段 (956):** Inverse orthogonal transformation unit.
- 量子化手段 (957):** Quantization unit.
- 逆量子化手段 (958):** Inverse quantization unit.
- 7フレーム内7フレーム間 (959):** 7-frame intra/inter-frame processing unit.
- 直交変換手段 (960):** Orthogonal transformation unit.
- 符号化制御手段 (961):** Encoding control unit.
- 量子化係数 (962):** Quantization coefficients.
- 可変長符号化手段 (963):** Variable length encoding unit.
- 出力 (964):** The final output of the system.
- 逆直交変換手段 (965):** Inverse orthogonal transformation unit.
- 量子化手段 (966):** Quantization unit.
- 逆量子化手段 (967):** Inverse quantization unit.
- 7フレーム内7フレーム間 (968):** 7-frame intra/inter-frame processing unit.
- 直交変換手段 (969):** Orthogonal transformation unit.
- 符号化制御手段 (970):** Encoding control unit.
- 量子化係数 (971):** Quantization coefficients.
- 可変長符号化手段 (972):** Variable length encoding unit.
- 出力 (973):** The final output of the system.
- 逆直交変換手段 (974):** Inverse orthogonal transformation unit.
- 量子化手段 (975):** Quantization unit.
- 逆量子化手段 (976):** Inverse quantization unit.
- 7フレーム内7フレーム間 (977):** 7-frame intra/inter-frame processing unit.
- 直交変換手段 (978):** Orthogonal transformation unit.
- 符号化制御手段 (979):** Encoding control unit.
- 量子化係数 (980):** Quantization coefficients.
- 可変長符号化手段 (981):** Variable length encoding unit.
- 出力 (982):** The final output of the system.
- 逆直交変換手段 (983):** Inverse orthogonal transformation unit.
- 量子化手段 (984):** Quantization unit.
- 逆量子化手段 (985):** Inverse quantization unit.
- 7フレーム内7フレーム間 (986):** 7-frame intra/inter-frame processing unit.
- 直交変換手段 (987):** Orthogonal transformation unit.
- 符号化制御手段 (988):** Encoding control unit.
- 量子化係数 (989):** Quantization coefficients.
- 可変長符号化手段 (990):** Variable length encoding unit.
- 出力 (991):** The final output of the system.
- 逆直交変換手段 (992):** Inverse orthogonal transformation unit.
- 量子化手段 (993):** Quantization unit.
- 逆量子化手段 (994):** Inverse quantization unit.
- 7フレーム内7フレーム間 (995):** 7-frame intra/inter-frame processing unit.
- 直交変換手段 (996):** Orthogonal transformation unit.
- 符号化制御手段 (997):** Encoding control unit.
- 量子化係数 (998):** Quantization coefficients.
- 可変長符号化手段 (999):** Variable length encoding unit.
- 出力 (1000):** The final output of the system.



【図 1 1】



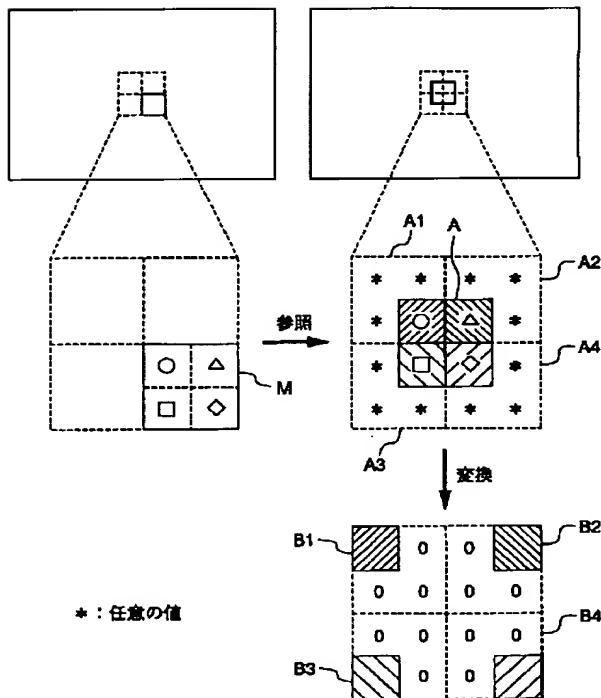
【図 1 3】



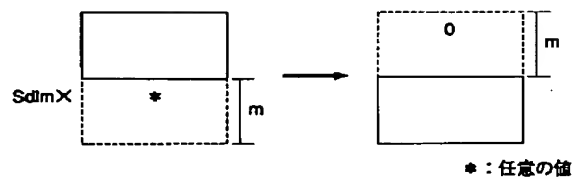
【図 1 2】

入力された画像データで動き補償を伴うフレーム間予測誤差に対して、直交変換を行うハイブリッド符号化されたデータを逆直交変換したデータのフレーム

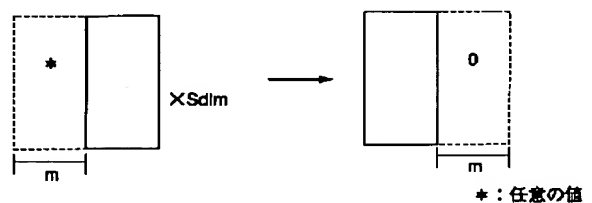
画像メモリに記憶されていた直交変換係数を逆直交変換したデータのフレーム



【図 1 5】

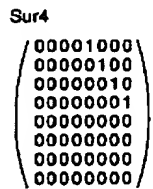
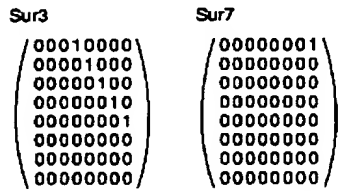
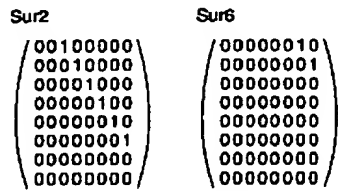
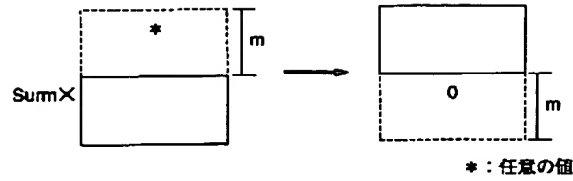
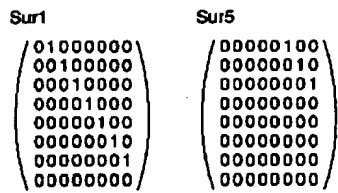


【図 1 6】

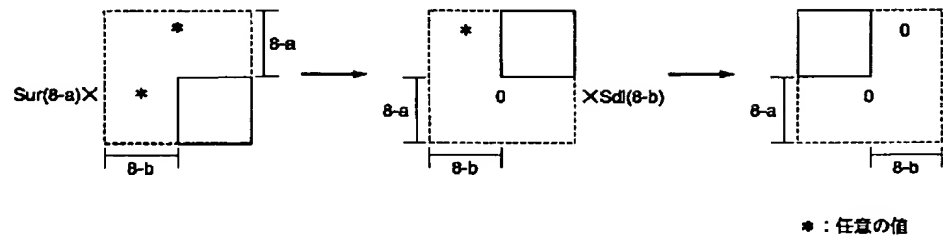




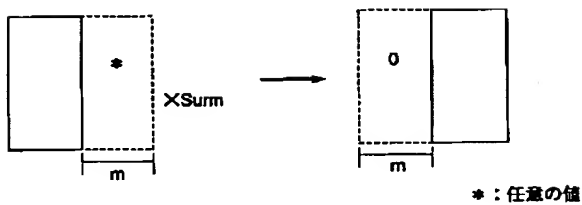
【図 14】



【図 19】



【図 18】



【図 20】

